ANL-77-12
Rev. 1

ANL-77-12
Rev. 1

DISPL: A SOFTWARE PACKAGE FOR ONE AND TWO SPATIALLY DIMENSIONED KINETICS-DIFFUSION PROBLEMS

by

G. K. Leaf, M. Minkoff, G. D. Byrne, D. Sorensen,
T. Bleakney, and J. Saltzman

U of C-AUA-USDOE

ARGONNE NATIONAL LABORATORY, ARGONNE, ILLINOIS

Prepared for the U. S. DEPARTMENT OF ENERGY
under Contract W-31-109-Eng-38

Printed in the United States of America
Available from
National Technical Information Service
U. S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22161
Price: Printed Copy $11.00; Microfiche $3.00

Distribution Category:
Mathematics and Computers
(UC-32)

ANL-77-12
Rev. 1

ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, Illinois 60439

DISPL: A SOFTWARE PACKAGE FOR ONE AND TWO SPATIALLY DIMENSIONED KINETICS-DIFFUSION PROBLEMS

by

G. K. Leaf, M. Minkoff, G. D. Byrne, D. Sorensen,
T. Bleakney, and J. Saltzman

Applied Mathematics Division

Revision 1: November 1978

(Supersedes ANL-77-12, May 1977)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

DISPL: A Software Package for One and Two Spatially Dimensioned Kinetics-Diffusion Problems

by

G. K. Leaf, M. Minkoff, G. D. Byrne, D. Sorensen,
T. Bleakney and J. Saltzman

ABSTRACT

DISPL is a software package for solving some second order nonlinear systems of partial differential equations including parabolic, elliptic, hyperbolic, and some mixed types such as parabolic-elliptic equations. Fairly general nonlinear boundary conditions are allowed as well as interface conditions for problems in an inhomogeneous media. The spatial domain is one or two dimensional with Cartesian, cylindrical, or spherical (in one dimension only) geometry. The numerical method is based on the use of Galerkin's procedure combined with the use of B-splines in order to reduce the system of PDE's to a system of ODE's. The latter system is then solved with a sophisticated ODE software package. Software features include extensive dump/restart facilities, free format input, moderate printed output capability, dynamic storage allocation, and three graphics packages.

Introduction

DISPL is a computer software system for solving a broad class of partial differential equations. This class includes problems which arise from the simulation of: time dependent heat conduction in nonhomogeneous media; the chemical kinetics-diffusion (or transport) of minor chemical species in the upper atmosphere; the steady state behavior of the boundary layer of a gas bubble immersed in a moving fluid; a water hammer; tertiary recovery of oil by thermal methods; steady state heat conduction; and chemical concentrations in both steady and unsteady tubular chemical reactors (packed or empty). This class then includes certain systems of partial differential equations of the following types: parabolic, elliptic, hyperbolic and certain mixed systems of these three types. Fairly general nonlinear boundary conditions are allowed and include those for which the solution, its flux or combinations of these are specified at the boundary, in perhaps a very complicated functional relationship. The basic set of equations consists of a substantial subset of

of the class of second order nonlinear systems of partial differential equations in one or two space-like variables and one time-like variable. The spatial domain is rectangular in either Cartesian, cylindrical or spherical (one dimension only) geometry. DISPL permits the presence of several material interfaces so that a variety of problems involving non-homogeneous media can be easily solved.

The numerical method used in DISPL can be described as follows. The system of partial differential equations is discretized in the space-like variables. This discretization is achieved by using a Galerkin procedure in conjunction with B-splines of a specified order and smoothness. This leads to a system of first order ordinary differential equations (ODE's) in the time-like variable which is solved by a sophisticated ODE software package. Currently, the ODE solver is a variation of GEAR.

The software aspects in DISPL of interest to the user are as follows. Input is via namelist which implies a free field format. An extensive dump and restart facility is available which has the following features. During either a steady-state or a transient calculation, a dump occurs automatically when the computer time allotted for the job has been exceeded. A dump can also be made at the normal conclusion of a steady-state calculation for use in starting a subsequent transient calculation. Furthermore, when time is exceeded, the dump is made in such a way that the internal routines can be restarted without any significant numerical effect. The restart is effected via a four-card change in the input deck. There are also extensive output capabilities including printed output and three graphics packages (one-dimensional slices, contours, and three-dimensional perspectives). Further the coding, other than that in the B-spline and ODE packages, and the user routines, is in MORTRAN. This language is a FORTRAN preprocessor which uses macro-instructions, adjusts the dimension of internal arrays at compile time, and allows structured programming.

In the sequel, we describe the problem types which can be solved by DISPL in §1. In §2, we describe the numerical methods used in this code. The package is treated in §§3,4,5,6, while sample problem coding and input are given in §7. Thus §§1 and 2 describe the underlying methods and strategies used in DISPL, while §§3-7 constitute a user's guide with examples.

To solve a given problem, the following two-tier approach to using this report is recommended.

I. Initial Use of the Package

1. Study §§ 1 and 3. This will provide a basic introduction to the form of the equation and terminology.
2. Use a previous deck (there are two sample decks provided with the code) as a basis for the current deck. That is, rather than developing the user deck from scratch, modify a previous one.
3. Use § 4 to determine the form of the user routines.
4. Use the Machine Readable Documentation listing to determine Namelist data.

II. Further Use of the Package

1. Refer to § 5 for assistance in modifying the macro variables, using the restart features, and interpreting error messages.
2. Refer to § 6 for interpretation of printed and graphical output.
3. Refer to § 7 for further examples.
4. Users interested in the mathematical and numerical procedures used should study § 2.

Finally, some cautionary remarks concerning this code. First, the code is designed as an engineering tool for use on a reasonably large class of practical problems. The code is not designed to replace either special purpose programs or production codes. Second, this code will not solve every second-order partial differential equation. Some of the more obvious restrictions include such aspects as rectangular spatial domains, no cross-derivatives, any side is of one boundary condition type, and second derivative information is not available for use in boundary conditions. Third, this program is primarily designed for nonlinear systems of parabolic equations. However, the program can solve some problems from other classes of PDE's including elliptic, hyperbolic, and mixed type problems such as parabolic-elliptic types (cf. § 7). This generality means that the user can specify ill-posed problems either through a conceptual error or an input error. Moreover, for some hyperbolic problems, the numerical method used in this code (Galerkin combined with a stiff ODE solver) will not be stable. Thus when using this code, the user has an obligation to see that he has formulated a meaningful problem for which a solution exists.

1. NATURE OF THE PARTIAL DIFFERENTIAL EQUATIONS

This program is designed to approximate the solution of a class of nonlinear parabolic systems of partial differential equations in one dimension (Cartesian, cylindrical, or spherical geometry) or two dimensions (Cartesian or cylindrical geometry). The approximation is based on the use of a Galerkin procedure to reduce the system of partial differential equations to a system of ordinary differential equations. This system of ordinary differential equations is then solved by means of a variant of the ANL version of the GEAR code [1]. In the Galerkin procedure, the class of approximating functions is generated from a tensor product basis of one-dimensional B-splines generated by subroutines developed by de Boor [2].

The starting point for this work was a desire to provide the capability for solving some simplified forms of the equation of continuity for a multicomponent chemically reactive fluid. Of course, in most fluid dynamics problems involving chemical reactions, it is necessary to carry out the simultaneous solution of the coupled equations of mass, momentum, and energy. However, there is a substantial class of problems for which a solution of the equation of continuity will suffice; it is for some of these problems that this code is intended.

The system of PDE's treated in this program does not have to be considered in any particular context; however, it may be helpful to use a physical model in order to provide some motivation for the choice of the class of PDE's treated here. To this end, consider a multicomponent fluid of M species with mass concentration ρ_i for the i-th species, and total mass concentration $\rho = \sum_{i=1}^{M} \rho_i$. Let $\vec{V}_i$ denote the velocity of the i-th species with respect to fixed coordinate axes, and $\vec{V} = \sum_{i=1}^{M} \rho_i \vec{V}_i / \rho$ denotes the local mass averaged velocity. Let q_i denote the rate of production of the i-th species from chemical reactions and other volume distributed sources, and $\vec{J}_i = (\vec{V}_i - \vec{V})\rho_i$ denote the mass flux relative to the mass averaged velocity. The equation of continuity for the i-th species is then given by

$$\frac{\partial \rho_i}{\partial t} + \nabla\cdot(\rho_i \vec{V} + \vec{J}_i) = q_i, \qquad 1 \le i \le M \tag{1.1}$$

where $\nabla\cdot$ denotes the divergence operator. If the flux $\vec{J}_i$ can be approximated

by

$$\vec{J}_i = -\rho \mathcal{D}_i \nabla(\rho_i/\rho) \quad \text{(Fick's law of diffusivity)}$$

then we have

$$\frac{\partial \rho_i}{\partial t} + \nabla\cdot(\rho_i \vec{V}) = \nabla\cdot(\rho \mathcal{D}_i \nabla(\rho_i/\rho)) + q_i, \qquad 1 \le i \le M . \tag{1.2}$$

If we assume that the total mass density ρ is constant, then $\nabla\cdot(\rho \mathcal{D}_i \nabla(\rho_i/\rho)) = \nabla\cdot(\mathcal{D}_i \nabla \rho_i)$. Moreover, since $\sum_{i=1}^{M} \vec{J}_i = \sum_{i}^{M} q_i = 0$, we find by summing Eq. (1.2) $\frac{\partial \rho}{\partial t} + \nabla\cdot(\rho \vec{V}) = 0$. When ρ is constant this reduces to $\nabla\cdot\vec{V} = 0$. Now $\nabla\cdot(\rho_i \vec{V}) = \vec{V}\cdot\nabla\rho_i + \rho_i \nabla\cdot\vec{V}$; thus when ρ is constant, Eq. (1.2) reduces to

$$\frac{\partial \rho_i}{\partial t} + \vec{V}\cdot\nabla\rho_i = \nabla\cdot(\mathcal{D}_i \nabla \rho_i) + q_i, \qquad 1 \le i \le M \tag{1.3}$$

or since $\nabla\cdot\vec{V} = 0$ we can write this equation in its conservative form.

$$\frac{\partial \rho_i}{\partial t} + \nabla\cdot(\rho_i \vec{V}) = \nabla\cdot(\mathcal{D}_i \nabla \rho_i) + q_i, \qquad 1 \le i \le M . \tag{1.4}$$

Equations (1.3) and (1.4) provide a basis for the selection of a class of PDE's treated by the program. With this physical model in mind, we consider the following system of nonlinear parabolic equations.

$$\begin{aligned} [\rho C_p]_m(t,r,z,\vec{u}) \frac{\partial u_m}{\partial t} &+ \theta\nabla\cdot(\vec{V}_m(t,r,z,\vec{u})u_m) + (1-\theta)\vec{V}_m(t,r,z,\vec{u})\cdot\nabla u_m \\ &= \nabla\cdot(\vec{\mathcal{D}}_m(t,r,z,\vec{u},\vec{\nabla u})\nabla u_m) + \sum_{m'=1}^{M} c_{mm'} u_{m'} + \\ &\sum_{m'=1}^{M}\sum_{m''=1}^{M} c_{mm'm''} u_{m'} u_{m''} + f_m(t,r,z,\vec{u},\vec{\nabla u}) \text{ for } 1 \le m \le M . \end{aligned} \tag{1.5}$$

Here $u_m = u_m(r,z,t)$ denotes the dependent variable (e.g. concentration of the m-th species), and we have included the option of considering a system in the substantial derivative form ($\theta = 0$) or in the conservative form ($\theta = 1$). The mass averaged velocity has been somewhat generalized by the inclusion of a two component $\vec{V}_m = (V_m^r, V_m^z)^T(t,r,z,\vec{u})^\dagger$ user supplied function which, as

---

†Here T denotes the transpose.

indicated, may depend on the species index m as well as (t,r,z) and u(t,r,z) = $(u_1,u_2,\ldots,u_M)^T(t,r,z)$. The coefficient of diffusivity $\vec{\mathcal{D}}_m = (\mathcal{D}_m^r,\mathcal{D}_m^z)^T$ is a two component user supplied function with the indicated dependencies. The expression $(\vec{\mathcal{D}}_m \nabla u_m)$ is interpreted as the two component vector

$$\left(\mathcal{D}_m^r \frac{\partial u_m}{\partial r},\ \mathcal{D}_m^z \frac{\partial u_m}{\partial z}\right)^T$$

so that anisotropic diffusion can be taken into account. The heat capacity coefficient $[\rho C_p]_m(t,r,z,u)$ is included for heat conduction problems when n = 1 and u(t,r,z) represents the temperature. Note, however, that this system allows ρC_p to depend on the species index m when M > 1; in particular $[\rho C_p]_m$ can be identically zero. As indicated, first and second order reaction rate coefficients are assumed to be constant. These rate constants are provided to the program through the input. The distributed source $f_m(t,r,z,u,\nabla u)$ is user supplied, and it can have the indicated dependencies when $\vec{u} = (u_1,\ldots,u_M)^T(t,r,z)$ and $\overrightarrow{\nabla u} = (\nabla u_1,\ldots,\nabla u_M)^T(t,r,z)$. The convection velocity $\vec{V}_m(t,r,z,\vec{u})$, diffusivity $\vec{\mathcal{D}}_m(t,r,z,\vec{u},\overrightarrow{\nabla u})$, heat capacity $[\rho C_p](t,r,z,\vec{u})$, and the distributed source $f_m(t,r,z,\vec{u},\overrightarrow{\nabla u})$ are made available to the program by means of user supplied subroutines (VEL, DIFUSE, RHOCP, and EXTSRC, respectively).

1.1 Domain of the Equation

Equation (1.5) is considered over a domain $R = [\underline{R},\overline{R}] \times [\underline{Z},\overline{Z}]$ which is rectangular with sides parallel to the coordinate axes. The geometry can be either Cartesian, cylindrical or spherical. In order to allow for diffusion in an inhomogeneous medium, the domain R can be composed of subrectangles where each subrectangle can have its own material properties. It is assumed that these subrectangles are formed by a set of NTIR vertical interfaces and a set of NTIZ horizontal interfaces as shown in Fig. 1.

Here the four sides of the domain R are indexed from 1 to 4 counterclockwise starting on the left hand side of R as indicated by the symbols $\boxed{1},\ldots,\boxed{4}$ in Fig. 1. This indexing scheme for the sides is used throughout this report.

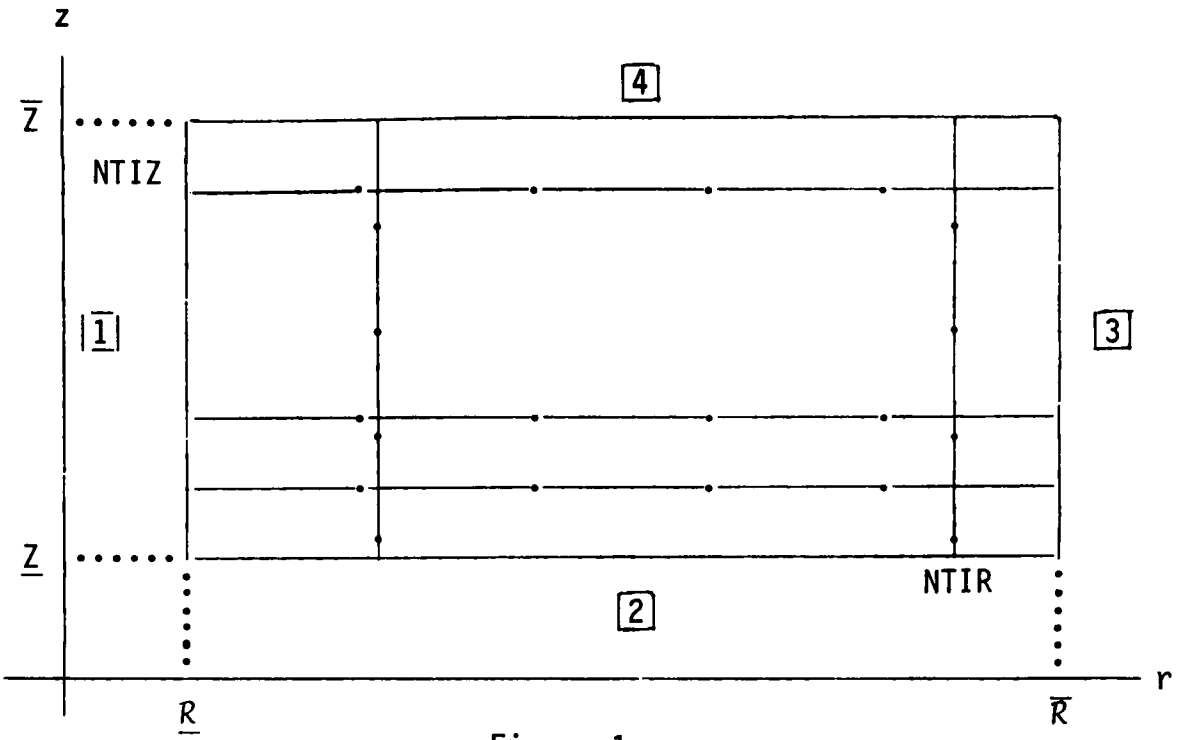


Figure 1

Interface Partition of a Domain

1.2 Boundary Conditions

This code does not require boundary conditions to be imposed on each of the four sides for each species. Thus, in principle, this code can deal with a larger class of problems than the set of coupled nonlinear parabolic equations. However, this larger class includes problems which are not well posed as well as problems for which the numerical method used in this code is unstable. For this reason, the discussion is primarily concerned with the class of parabolic type problems.

For each species m, a boundary condition of the following form may be specified on each of the four sides of the domain R.

$$\alpha h u_m + \beta \vec{D}_m \nabla u_m \cdot \vec{n} = \gamma h \rho_m^0 \tag{1.6}$$

where $\alpha = \alpha(m,s)$, $\beta = \beta(m,s)$, $\gamma = \gamma(m,s)$, $1 \le m \le M$, $1 \le s \le 4$ are specified constants which depend on the species index m and the side index s. On the left hand side of R (side 1), we have $h = h_1(J,m)$, $1 \le J \le NTIZ+1$, $1 \le m \le M$; and on side 3, we have $h = h_3(J,m)$, $1 \le J \le NTIZ+1$, $1 \le m \le M$. On sides 2 and 4, we have $h = h_2(I,m)$ and $h = h_4(I,m)$ respectively for $1 \le I \le NTIR+1$,

$1 \le m \le M$. Recall that NTIR(NTIZ) is the number of vertical (horizontal) interfaces in R; so that for each species m, these mass transfer coefficients h can depend on the materials which are present on the boundary of R. The function $\rho_m^0 = \rho^0(t,m,s,x)$ is specified by the user as functions of time t, species m, side s, and the position x on side s (thus, if s = 1 or 3, x = z and if s = 2 or 4, x = 4). In addition, the function ρ^0 can depend on $\vec{u}$ and the derivative $\nabla u \cdot |\vec{n}|$, each evaluated at x on side s. Thus the code can handle nonlinear boundary conditions as well as boundary conditions involving relationships between the species at the boundary. Here $\vec{n} = \vec{n}(s)$ denotes the exterior unit normal for the boundary ∂R of R. Thus if $\vec{r}$ and $\vec{z}$ are the unit coordinate vectors, we have $\vec{n}(1) = -\vec{r}$, $\vec{n}(2) = -\vec{z}$, $\vec{n}(3) = \vec{r}$, and $\vec{n}(4) = \vec{z}$. In addition, we allow for the possibility that ρ^0 may depend on the sign of the normal component $\vec{V}_m \cdot \vec{n}$ of the convection velocity on the boundary, e.g. $\rho^0 = 0$ if $\vec{V} \cdot \vec{n} < 0$.

Since $\vec{V}_m \cdot \vec{n}$ is available, boundary conditions of the form

$$\alpha h u_m - \beta(u_m \vec{V}_m - \mathcal{D}_m \nabla u_m) \cdot \vec{n} = \gamma h \rho_m^0$$

can be treated in this code by writing this condition in the form

$$\alpha h u_m + \beta \vec{\mathcal{D}}_m \nabla u_m \cdot \vec{u} = \gamma h \rho_m^0 + \beta u_m \vec{V}_m \cdot \vec{n}$$

and supplying the r.h.s. in subroutine BRHØ. The motivation for this type of condition is provided by the mass transport model where $(u_m \vec{V}_m - \vec{\mathcal{D}}_m \nabla u_m) \cdot \vec{n}$ is the normal component of the total mass flux.

1.3 Interface Conditions

Let NTIR denote the number of vertical interfaces and NTIZ the number of horizontal interfaces in R as shown in Fig. 1. Note that an interface extends from one external boundary of R to the opposite external boundary. Let Γ denote any one of these interfaces and let $|\vec{n}|$ denote the positive unit vector $\hat{r}$ if Γ is vertical and the positive unit vector $\hat{z}$ if Γ is horizontal. If $F(t,r,z)$ is any given function, $F|_{\Gamma^-}$ will denote the limiting value of F on Γ

from the left (below) if Γ is vertical (horizontal) while $F|_{\Gamma^+}$ denotes the limiting value from the right (above) if Γ is vertical (horizontal).

This program allows for two possible types of interface conditions to be imposed on u_m at an interface Γ.

A. Continuous Interface Condition

Before stating this interface condition, we impose the following restriction on the behavior of $\vec{V}_m$ across an interface.

(H1.1) $\vec{V}_m(t,r,z,\vec{u})$ as a function of (r,z) is continuous across an interface Γ.

The continuous interface condition on Γ requires that the concentration and the normal component of the total flux be continuous across the interface Γ, that is

$$u_m(t,r,z)|_{\Gamma^-} = u_m(t,r,z)|_{\Gamma^+} \tag{1.7}$$

$$(\vec{V}_m u_m - \vec{\mathcal{D}}_m \nabla u_m)\cdot|\vec{n}|\,|_{\Gamma^-} = (\vec{V}_m u_m - \vec{\mathcal{D}}_m \nabla u_m)\cdot|\vec{n}|\,|_{\Gamma^+} \quad .$$

Since $\vec{V}_m$ is assumed to be continuous across Γ, this last condition reduces to

$$\vec{\mathcal{D}}_m \nabla u_m \cdot|\vec{n}|\,|_{\Gamma^-} = \vec{\mathcal{D}}_m \nabla u_m \cdot|\vec{n}|\,|_{\Gamma^+} \tag{1.8}$$

B. Gap Interface Condition

The gap interface condition is intended for use in problems of heat transfer between solids of different material properties. For this reason it is assumed in this case that $\vec{V}_m \equiv 0$, and the conditions are of the following form.

$$\text{a)}\quad -\vec{\mathcal{D}}_m \nabla u_m \cdot|\vec{n}|\,|_{\Gamma^-} = h^g\{u_m|_{\Gamma^-} - u_m|_{\Gamma^+}\}$$

(1.9)

$$\text{b)}\quad \vec{\mathcal{D}}_m \nabla u_m \cdot|\vec{n}|\,|_{\Gamma^-} = \vec{\mathcal{D}}_m \nabla u_m \cdot|\vec{n}|\,|_{\Gamma^+}$$

Note that if Γ is a vertical interface, then $\vec{\mathcal{D}}_m \nabla u_m \cdot|\vec{n}| = \mathcal{D}_m^r \frac{\partial u_m}{\partial r}$; while

$\vec{v}_m \nabla u_m \cdot |\vec{n}| = v_m^z \frac{\partial u_m}{\partial z}$ if Γ is a horizontal interface. The gap interface is intended primarily for problems in heat transfer when the number of species M is equal to one. Although the condition (1.9) can be used when $M > 1$, the program imposes the following restriction on gap interfaces.

(H1.2) If Γ is a gap interface for one species then it must be a gap interface for all species.

Thus the classification of interfaces into two types -- continuity and gap -- is independent of species.

The gap coefficients h^g can depend on the following parameters. First consider the set of all vertical gap interfaces and let this set be indexed by I, $1 \le I \le NIGAP \le NTIR$. The I-th vertical gap will intersect a set of NTIZ horizontal interfaces. This horizontal set will subdivide the vertical gap into a set of NTIZ+1 subintervals. The vertical gap coefficients can depend on the following parameters.

(1.10) $h^{Vg} = h^{Vg}(m,I,J)$ where $1 \le m \le M$, $1 \le I \le NIGAP$, and $1 \le J \le NTIZ+1$.

In the same way, if NJGAP denotes the number of horizontal gaps, then the horizontal gap coefficients can depend on the following parameters.

(1.11) $h^{Hg} = h^{Hg}(m,J,I)$ where $1 \le m \le M$, $1 \le J \le NJGAP$, and $1 \le I \le NTIR+1$.

1.4 Initial Conditions

The system (1.5) - (1.9) will be complete when a set of initial conditions are specified. This program allows for two possibilities. First, an arbitrary initial distribution $\{u_m^0(r,z): 1 \le m \le M\}$ can be specified by the user. The second possible approach is to start from a steady-state or equilibrium distribution $\{\tilde{u}_m^0(r,z): 1 \le m \le M\}$. The program computes this steady-state distribution (prior to the start of the transient calculation) as the asymptotic solution (i.e., as the time t goes to infinity) corresponding to specified time-independent convection velocities $\vec{V}_m$, diffusion

coefficients $\vec{\mathcal{D}}_m$, heat capacity coefficients $[\rho C_p]_m$ distributed sources f_m, and external boundary source ρ_m^0. Since the program calculates the steady-state solution as an asymptotic solution in time of a system of first order ODE's, the program requires an initial estimate for the steady-state solution. The program allows for two possibilities. First, the user may provide an initial estimate for the steady-state distribution just as he would provide an initial distribution for a transient calculation. The second option is to let the program generate an initial estimate for the steady-state solution.

In summary, this program approximates the solution to the nonlinear parabolic system (1.5), (1.6), (1.8), and (1.9) over a two-dimensional rectangular domain in either Cartesian, cylindrical, or spherical (one dimensional) geometry. The domain may be subdivided into subrectangles having different material properties with specified interface conditions. The user supplies the convection velocities $\vec{V}_m(t,r,z,\vec{u})$, the diffusion coefficient $\vec{\mathcal{D}}_m(t,r,z,\vec{u},\overrightarrow{\nabla u})$, the heat capacity coefficients $[\rho C_p]_m(t,r,z,\vec{u})$, the distributed sources $f_m(t,r,z,\vec{u},\overrightarrow{\nabla u})$, and the external boundary source $\rho^0(t,m,s,x)$. First and second order constant reaction rates can be specified on input. Higher order and variable reaction rates are specified in the distributed sources. The program can perform either a transient calculation, or a steady-state calculation, or a steady-state calculation followed by a transient calculation. The program also has a restart capability which will be described in a later section.

2. APPROXIMATION PROCEDURE

This program approximates the solution to Eqs. (1.5) - (1.9) by means of a Galerkin type procedure based on the use of a tensor product basis of one-dimensional B-splines. The integrals which arise from the Galerkin procedure are evaluated by means of product Gauss-Legendre quadrature formulas.

2.1 Weak Form of the Equation

The Galerkin procedure used in this program starts from a weak form of Eqs. (1.5), (1.6), (1.8), and (1.9). In order to simplify the notation, we will drop the species index m in this discussion. For each species, Eq. (1.5) has the general form

$$(2.1) \qquad [\rho C_p]\,\frac{\partial u}{\partial t} = -\theta\nabla\cdot(\vec{V}u) - (1-\theta)\vec{V}\cdot\nabla u + \nabla\cdot(\vec{\mathcal{D}}\nabla u) + F(t,r,z,u,\nabla u)\ ,$$

where $\rho C_p = \rho C_p(t,r,z,u)$, $\vec{V} = \vec{V}(t,r,z,u)$ and $\vec{\mathcal{D}} = \vec{\mathcal{D}}(t,r,z,u)$. Here we have grouped the reaction sources and the distributed source into the general term F. Recall that the rectangular domain R is the union of material subrectangles $\{R_s\}$ defined by a set of vertical and horizontal interfaces. Let $\omega = \omega(r,z)$ be any function belonging to the class $C^1(R_s)$ for each s and, for the moment, satisfying no other constraints. To obtain a weak form, we multiply Eq. (2.1) by ω and integrate the resulting equation over the domain R. If $\langle u,\omega\rangle = \iint_R u\omega$, we obtain the expression

$$(2.2) \qquad \langle[\rho C_p]\,\frac{\partial u}{\partial t},\ \omega\rangle = -(1-\theta)\langle\vec{V}\cdot\nabla u,\omega\rangle + \langle\nabla\cdot(\vec{\mathcal{D}}\nabla u-\theta\vec{V}u),\omega\rangle + \langle F,\omega\rangle\ .$$

To obtain the weak solution, we first apply Green's theorem to the divergence term over each subrectangle R_s. Before doing this we will introduce some notation for boundaries and interfaces. Let $\partial\tilde{R}_1$ denote that part of the external boundary ∂R for which $\beta \neq 0$ in the boundary condition (1.6), let $\partial\tilde{R}_2$ denote that part of the external boundary where no boundary condition is imposed, and let ∂R_0 denote that part where $\beta = 0$ in condition (1.6). Thus $\partial R = \partial R_0 \cup \partial\tilde{R}_1 \cup \partial\tilde{R}_2$, and each part ∂R_0, $\partial\tilde{R}_1$, and $\partial\tilde{R}_2$ may be different for different species. For the moment we set $\partial\tilde{R} = \partial\tilde{R}_1 \cup \partial\tilde{R}_2$. Boundary conditions on ∂R_0 are usually referred to as essential conditions. If ∂R_s denotes the

boundary of a material subrectangle R_s, then let Γ_s denote that part of ∂R_s which is an interface, that is Γ_s is not part of the external boundary ∂R. Now $\Gamma_s = \Gamma_s^g \cup \Gamma_s^c$ where Γ_s^g is that part of Γ_s which is a gap interface and Γ_s^c is that part which is a continuity interface.

Consider the divergence term in Eq. (2.2) and apply Green's theorem over each subrectangle R_s.

$$(2.3)\quad \langle \nabla\cdot(\mathcal{D}\nabla u-\theta\vec{V}u),\omega\rangle = \sum_s \iint_{R_s} \omega\nabla\cdot(\mathcal{D}\nabla u-\theta\vec{V}u)$$

$$= \sum_s \int_{\Gamma_s} \omega(\mathcal{D}\nabla u-\theta\vec{V}u)\cdot\vec{n}_s + \int_{\partial\tilde{R}} \omega(\mathcal{D}\nabla u-\theta\vec{V}u)\cdot\vec{n} + \int_{\partial R_0} \omega(\mathcal{D}\nabla u-\theta\vec{V}u)\cdot\vec{n}$$

$$- \iint_R (\mathcal{D}\nabla u-\theta\vec{V}u)\cdot\nabla\omega \ .$$

For each Γ_s, the integral over Γ_s appears twice in the above sum with the direction of integration reversed; therefore, terms of the following form will appear in the above sum.

$$(2.4)\quad \int_{|\Gamma_s|} \omega^-(\mathcal{D}\nabla u-\theta\vec{V}u)^-\cdot|\vec{n}_s| - \omega^+(\mathcal{D}\nabla u-\theta\vec{V}u)^+\cdot|\vec{n}_s| \ .$$

Here $|\vec{n}_s|$ denotes the unit normal to the interface Γ_s oriented in the positive coordinate direction; thus $(\mathcal{D}\nabla u-\theta\vec{V}u)\cdot|\vec{n}_s| = \mathcal{D}^r \frac{\partial u}{\partial r} - \theta V^r u$ if Γ_s is a vertical interface and $(\mathcal{D}\nabla u-\theta\vec{V}u)\cdot|n_s| = \mathcal{D}^z \frac{\partial u}{\partial z} - \theta V^z u$ if Γ_s is a horizontal interface. The symbol $\int_{|\Gamma_s|}$ indicates that the integral is taken in the positive coordinate direction. As usual, ω^- and ω^+ indicate limiting values on Γ_s taken from the left (below) and from the right (above) respectively if Γ_s is a vertical (horizontal) interface. Now $\Gamma_s = \Gamma_s^c \cup \Gamma_s^g$, so consider a continuous interface Γ_s^c. If $\theta = 1$, then condition (1.8) states that

$$(\mathcal{D}\nabla u-\vec{V}u)^-\cdot|\vec{n}_s| = (\mathcal{D}\nabla u-\vec{V}u)^+\cdot|\vec{n}_s| \ .$$

Thus in either case $\theta = 0$ or 1, we find that the integral in (2.4) has the following form on Γ_s^c.

$$(2.5) \qquad \int_{|\Gamma_s^c|} (\omega^- - \omega^+)(\mathcal{D}\nabla u - \theta\vec{V}u)^- \cdot |\vec{n}_s|$$

Hence if we assume, as we shall, that ω is continuous across each continuity interface Γ_s^c, then these integrals over Γ_s^c will not contribute to the sum appearing in Eq. (2.3).

Next we consider a gap interface Γ_s^g. At such an interface we have assumed that $\vec{V} = 0$ or at least that $\vec{V}\cdot|\vec{n}_s| = 0$; hence using conditions (1.9), we find that integral (2.4) on Γ_s^g has the following form.

$$(2.6) \qquad -\int_{|\Gamma_s^g|} (\omega^- - \omega^+)h^g(u^- - u^+) \ .$$

Next, we consider the integral over $\partial\tilde{R} = \partial\tilde{R}_1 \cup \partial\tilde{R}_2$ appearing in Eq. (2.3). On $\partial\tilde{R}_1$, $\beta \neq 0$; thus we can use the boundary condition (1.6) to eliminate $\mathcal{D}\nabla u$. From (1.6), we have

$$(2.7) \qquad \mathcal{D}\vec{\nabla}u\cdot\vec{n} = -\beta^{-1}h(\alpha u - \gamma\rho^0) \ ,$$

which gives

$$(2.8) \qquad \int_{\partial\tilde{R}} \omega(\mathcal{D}\nabla u - \theta u\vec{V})\cdot\vec{n} = \int_{\partial\tilde{R}_1} \omega[(-\theta)u\vec{V}\cdot\vec{n} - \beta^{-1}h(\alpha u - \gamma\rho^0)] + \int_{\partial\tilde{R}_2} \omega[\mathcal{D}\nabla u - \theta u\vec{V}]\cdot\vec{n} \ .$$

Next we consider the integral over ∂R_0 appearing in Eq. (2.3). On ∂R_0 we have $\beta = 0$, so that $u = \gamma\rho^0/\alpha$ on ∂R_0. Since this form does not help in simplifying the integral appearing in Eq. (2.3), we will restrict the class of test functions from which ω is selected by requiring that $\omega \equiv 0$ on ∂R_0. Then this integral will not contribute to Eq. (2.3). The essential condition $u = \gamma\rho^0/\alpha$ will then be applied in a weak form by requiring

$$(2.9) \qquad \int_{\partial R_0} \omega u = \int_{\partial R_0} \frac{\gamma}{\alpha}\rho^0\omega \ .$$

From (2.6) and (2.8), we find that Eq. (2.3) takes the form

$$(2.10)\qquad \langle \nabla\cdot(\mathcal{D}\nabla u-\theta\vec{V}u),\omega\rangle = -\sum_s \int_{|\Gamma_s^g|} (\omega^- - \omega^+)h^g(u^- - u^+)$$
$$+ \int_{\partial\tilde{R}_1} \omega[(-\theta)u\vec{V}\cdot\vec{n} - \beta^{-1}h(\alpha u - \gamma\rho^0)] + \int_{\partial\tilde{R}_2} \omega[\mathcal{D}\nabla u - \theta u\vec{V}]\cdot\vec{n}$$
$$- \iint_R (\mathcal{D}\nabla u - \theta\vec{V}u)\cdot\nabla\omega \ .$$

Using this expression in Eq. (2.2), we find

$$(a)\quad \langle [\rho C_p]\frac{\partial u}{\partial t}, \omega\rangle = -(1-\theta)\langle \vec{V}\cdot\nabla u,\omega\rangle - \sum_s \int_{|\Gamma_s^g|} (\omega^- - \omega^+)h^g(u^- - u^+)$$
$$- \theta \int_{\partial\tilde{R}_1} \omega u\vec{V}\cdot\vec{n} - \beta^{-1}\int_{\partial\tilde{R}_1} \omega(\alpha h u - \gamma h\hat{\rho^0}) + \int_{\partial\tilde{R}_2} \omega[\vec{\mathcal{D}}\nabla u - \theta u\vec{V}]\cdot\vec{n}$$
$$- \iint_R (\mathcal{D}\nabla u - \theta u\vec{V})\cdot\nabla\omega + \iint_R F\omega \ ,$$

(2.11)

$$(b)\quad \int_{\partial R_0} \omega u = \frac{\gamma}{\alpha}\int_{\partial R_0} \rho^0\omega \ ,$$

$$(c)\quad \iint_R u(t=t_0)\omega = \iint_R u^0\omega \ ,$$

$$(d)\quad \int_{\partial R_0} \omega u(t=t_0) = \frac{\gamma}{\alpha}\int_{\partial R_0} \omega\rho^0(t=t_0) \ .$$

In Eq. (2.11a), ω is allowed to range over the set of all functions which are continuous together with their first derivatives in each material subrectangle R_s, and which are continuous across each continuous interface, and which vanish on the essential boundary ∂R_0. In Eq. (2.11b), ω is allowed to range over the set of functions ω defined on ∂R_0 which are continuous together with their first derivatives in each material subrectangle on ∂R_0 and which are continuous across every continuous interface on ∂R_0. In Eq. (2.11c), ω

ranges over the same set of functions as in Eq. (2.11a), and in Eq. (2.11d), ω ranges over the same set as in Eq. (2.11b). The system (2.11) represents the weak form to which the Galerkin approximation is applied.

2.2 Approximating Subspace

A Galerkin procedure seeks a solution in a subspace spanned by a given basis set. The basis sets used in this program are tensor product basis generated from one-dimensional B-splines. The B-splines are calculated using a subroutine package written by de Boor [2]. For completeness, we will describe B-splines in one dimension. This presentation is based on the work of de Boor in [2,3].

2.2.1 B-Spline Basis

Let an interval [a,b] be subdivided by a mesh

$$\pi: a = X_1 < X_2 <\ldots< X_\ell < X_{\ell+1} = b .$$

The points x_i will be referred to as breakpoints. In this subsection we develop a basis for spaces of functions which are piecewise polynomials over this mesh π. Let

$$\mathbb{P}_{k,\pi} = \{f(x): \text{in each interval } [X_i,X_{i+1}],\ 1 \le i \le \ell,\ f(x) \text{ is equal to a polynomial of order } k \text{ (degree at most } k-1)\} .$$

Since there are ℓ subintervals, it is clear that the dimension of $\mathbb{P}_{k,\pi}$ is $k\ell$. Observe that any function in $\mathbb{P}_{k,\pi}$ may be discontinuous at the interior breakpoints $\{x_i: 2 \le i \le \ell\}$. We now consider subspaces of $\mathbb{P}_{k,\pi}$ generated by imposing smoothness constraints on elements in $\mathbb{P}_{k,\pi}$ at these interior breakpoints. Let $\nu = (\nu_2,\ldots,\nu_\ell)$ be a set of specified integers with $0 \le \nu_i \le k-1$, and let $S_{k,\pi,\nu}$ denote the space of function $f(x)$ in $\mathbb{P}_{k,\pi}$ for which $f^{(j)}(X_i^-) = f^{(j)}(X_i^+)$, $0 \le j \le \nu_i-1$, for $2 \le i \le \ell$. At each breakpoint X_i we have imposed a set of ν_i constraints on a function in $\mathbb{P}_{k,\pi}$. Thus the dimension of the subspace $S_{k,\pi,\nu}$ will be

$$k\ell - \sum_{i=2}^{\ell} \nu_i = k + \sum_{i=2}^{\ell} (k-\nu_i) .$$

We now consider the construction of a basis for $S_{k,\pi,\nu}$ such that each element of the basis has local support; i.e., is non-zero on only a few subintervals, and furthermore each element is non-negative. The space $S_{k,\pi,\nu}$ is a space of polynomial splines and a basis of the above type is called a B-spline basis. To generate such a basis, we will consider divided differences of order k of the truncated power functions $g_k(\sigma;s) = (\sigma-s)_+^{k-1}$. To this end, let $\{\xi_i\}$ be any non-decreasing sequence of points (they will be referred to as knots) subject to the condition that $\xi_i < \xi_{i+k}$ for all i. Let $M_i(s)$ denote the k-th divided difference of $g_k(\sigma;s)$ with respect to the knots $\xi_i,\ldots,\xi_{i+k}$, that is $M_i(s) = g_k[\xi_i,\ldots,\xi_{i+k};s]$. Assuming for the moment that the knots $\xi_i,\ldots,\xi_{i+k}$ are distinct, then the divided difference $M_i(s)$ would have the representation

$$M_i(s) = \sum_{j=0}^{k} \frac{g_k(\xi_{i+j};s)}{\prod_{\substack{m=0 \\ m\neq j}}^{k} (\xi_{i+j}-\xi_{i+m})} .$$

If two of the knots $\xi_i,\ldots,\xi_{i+k}$ coincide, then the above representation will involve first derivatives of $g_k(\sigma;s)$ with respect to σ; if three knots coincide, then derivation of order up to two will appear, and so on. In general, if $\zeta_1,\ldots,\zeta_J$ are the distinct points among the knots $\{\xi_i\}$ and if each ζ_j appears d_j times, then $M_i(s)$ will be a linear combination of the functions $(\zeta_j-s)^{k-m}$, $1 \le m \le d_j$, $1 \le j \le J$. Note that if ν_j is the smoothness index of $M_i(s)$ at ζ_j ($M_i(s)$ has continuous derivatives at ζ_j through order ν_j-1), then $k = d_j+\nu_j$. It is easily seen that each $M_i(s)$ is a piecewise polynomial of order k with breakpoints $\zeta_1,\ldots,\zeta_J$ at which

$$M_i^{(m)}(\zeta_j^-) = M_i^{(m)}(\zeta_j^+) \quad \text{for} \quad 0 \le m \le k-1-d_j = \nu_j-1,\ 1 \le j \le J ,$$

$M_i(s)$ has its support in $[\xi_i,\xi_{i+k}]$, and each $M_i(s)$ is non-negative.

Now consider the subspace $S_{k,\pi,\nu}$. Define the set of knots $\{\xi_i:\ 1 \le i \le n+k\}$ where $n = k + \sum_{i=2}^{\ell} d_i$, $d_i = k-\nu_i$, and

$$\xi_1 = \cdots\cdots\cdots\cdots = \xi_k = X_1$$

$$\xi_{k+1} = \cdots\cdots\cdots\cdots = \xi_{k+d_2} = X_2$$

$$\vdots$$

$$\xi_{k+d_2+\ldots+d_{j-1}+1} = \cdots\cdots = \xi_{k+d_2+\ldots+d_j} = X_j, \quad j \leq \ell ,$$

and

$$\xi_{n+1} = \cdots\cdots\cdots\cdots = \xi_{n+k} = X_{\ell+1} .$$

That is, the first and last breakpoints have multiplicity k while the breakpoint X_j has multiplicity d_j; for $2 \leq j \leq \ell$. For each i, $1 \leq i \leq n$, we can form the divided difference $M_i(x)$, and the B-spline basis for $S_{k,\pi,\nu}$ is taken to be the so-called normalized B-splines $\{N_{i,k}(x): 1 \leq i \leq n\}$ where $N_{i,k}(x) = (\xi_{i+k}-\xi_i)M_i(x) = (\xi_{i+k}-\xi_i)g_k[\xi_i,\ldots,\xi_{i+k};x]$. Explicit representations of the normalized B-splines are generally rather involved due to possible multiplicity of knots; however the following properties are useful in applications of these functions.

(i) $\sum_{i=1}^{n} N_{i,k}(x) \equiv 1$ for $a \leq x \leq b$,

(ii) $N_{1,k}(X_1) = N_{n,k}(X_{\ell+1}) = 1$; $N_{i,k}(X_1) = 0$ for $i > 1$,

$N_{i,k}(X_{\ell+1}) = 0$ for $i < n$,

(iii) $N'_{1,k}(X_1) + N'_{2,k}(X_1) = 0$, $N'_{i,k}(X_1) = 0$ for $i > 2$,

(2.12)

$N'_{n-1,k}(X_{\ell+1}) + N'_{n,k}(X_{\ell+1}) = 0$, $N'_{i,k}(X_{\ell+1}) = 0$ for $i < n-1$,

$N'_{1,k}(X_1) = -\frac{(k-1)}{X_2-X_1}$, $N'_{n,k}(X_{\ell+1}) = \frac{(k-1)}{X_{\ell+1}-X_\ell}$,

(iv) $N_{i,k}(x) \equiv 0$ if $x \notin [\xi_i,\xi_{i+k}]$,

(v) $N_{i,k}(x) \geq 0$ for $a \leq x \leq b$.

2.2.2 Tensor Product Basis

We now consider the two-dimensional domain $R = [\underline{R},\overline{R}]\times[\underline{Z},\overline{Z}]$. Recall that this domain may be subdivided by a set of NTIR vertical interfaces and a set of NTIZ horizontal interfaces. In addition, we will allow for an additional set of NMR vertical mesh lines and an additional set of NMZ horizontal mesh lines. The set of interfaces and the set of additional mesh lines are interior to R; so that the interval $[\underline{R},\overline{R}]$ is subdivided by a set of NTIR+NMR interior (not including $\underline{R}$ or $\overline{R}$) points. Thus we have a partition

$$\pi_r: \underline{R} = r_1 < r_2 <\ldots< r_{\ell_r} < r_{\ell_r+1} = \overline{R}$$

of the interval $[\underline{R},\overline{R}]$ with $\ell_r = 1 + \text{NTIR} + \text{NMR}$. In the same way, we have a partition

$$\pi_z: \underline{Z} = z_1 < z_2 <\ldots< z_{\ell_Z} < z_{\ell_Z+1} = \overline{Z}$$

of the interval $[\underline{Z},\overline{Z}]$ with $\ell_z = 1 + \text{NTIZ} + \text{NMZ}$. Let k_r,k_z be integers greater than or equal to 2 specifying the order of the B-splines basis in the r variable and in the z variable, respectively. Consider the r variable first; we must specify a smoothness index $\nu^r = (\nu_2^r,\ldots,\nu_{\ell_r}^r)$. These indices are subject to the following restrictions:

(i) $0 \le \nu_\sigma^r \le k_r-1$,

(ii) if r_σ, corresponds to a vertical gap interface, then as we have seen, the solution can be discontinuous across this interface; hence we must have $\nu_{\sigma,}^r = 0$,

(iii) if r_σ, corresponds to a continuity interface, then as we have seen, the solution is continuous across this interface but the first derivative may be discontinuous across this interface; hence we must have $\nu_{\sigma,}^r = 1$.

A similar set of restrictions is placed on the horizontal smoothness indices $\nu_0^z = (\nu_2^z,\ldots,\nu_{\ell_z}^z)$. Let $\{A_i(r): 1 \le i \le N_r\}$ be the set of normalized B-splines

relative to k_r, π_r, and ν^r; and let $\{B_j(z): 1 \le j \le N_z\}$ be the set of normalized B-splines relative to k_z, π_z, and ν^z. Here

$$N_r = k_r + \sum_{\sigma=2}^{\ell_r} (k_r - \nu_\sigma^r) \quad \text{and} \quad N_z = k_z + \sum_{\tau=2}^{\ell_z} (k_z - \nu_\tau^z) .$$

Let the four sides of the domain R be numbered consecutively from 1 to 4 in the counterclockwise direction starting on the left side. Set

$$s_i = \begin{cases} 1 & \text{if side } i \text{ has an essential boundary condition } (\beta = 0 \text{ in Eq. 1.2}) \\ 0 & \text{if side } i \text{ has a non-essential boundary condition } (\beta \neq 0 \text{ in Eq. 1.2}) \end{cases}$$

$$\hat{n}_r = 1 + s_1 = \begin{cases} 2 & \text{if side 1 has an essential condition,} \\ 1 & \text{otherwise.} \end{cases}$$

$$\hat{N}_r = N_r - s_3 = \begin{cases} N_r - 1 & \text{if side 3 has an essential condition,} \\ N_r & \text{otherwise.} \end{cases}$$

$$\hat{n}_z = 1 + s_2,$$

$$\hat{N}_z = N_z - s_4.$$

For each species m, we seek an approximate solution of the following form

$$\begin{aligned} u_m(r,z,t) &= \sum_{i'=1}^{N_r} \sum_{j'=1}^{N_z} U_{i',j'}^m(t)\, A_{i'}(r)\, B_{j'}(z) \\ &= s_1 \sum_{j'=\hat{n}_z}^{\hat{N}_z} U_{1,j'}^m(t)\, A_1(r)\, B_{j'}(z) + s_2 \sum_{i'=1}^{N_r} U_{i',1}^m(t)\, A_{i'}(r)\, B_1(z) \\ &\quad + s_3 \sum_{j'=\hat{n}_z}^{\hat{N}_z} U_{N_r,j'}^m(t)\, A_{N_r}(r)\, B_{j'}(z) + s_4 \sum_{i'=1}^{N_r} U_{i',N_z}^m(t)\, A_{i'}(r)\, B_{N_z}(z) \\ &\quad + \sum_{i'=\hat{n}_r}^{\hat{N}_r} \sum_{j'=\hat{n}_z}^{\hat{N}_z} U_{i',j'}^m(t)\, A_{i'}(r)\, B_{j'}(z) . \end{aligned} \tag{2.13}$$

In the second expression, we have grouped the basis functions by identifying

the set which is associated with the essential boundary ∂R_0. That is, the set

$$(2.14) \quad T_0 = \{s_1 A_1(r) B_{j'}(z),\ s_2 A_{i'}(r) B_1(z),\ s_3 A_{N_r}(r) B_{j'}(z),\ s_4 A_{i'}(r) B_{N_z}(z):$$
$$1 \le i' \le N_r,\ \hat{n}_z \le j' \le \hat{N}_z\}$$

when restricted to ∂R_0 provides a basis for functions defined on ∂R_0. Moreover, the set

$$(2.15) \quad T = \{A_{i'}(r) B_{j'}(z):\ \hat{n}_r \le i' \le \hat{N}_r,\ \hat{n}_z \le j' \le \hat{N}_z\}$$

will provide a basis for functions defined on R which vanish on ∂R_0.

2.3 Galerkin Approximation

Given the basis sets T and T_0 defined in the last subsection, the Galerkin approximation applied to the system (2.11) proceeds as follows. For each species m, we seek an approximation to the concentration $u_m(t,r,z)$ of the form given in Eq. (2.13). Each species m has a weak equation of the form (2.11); thus for each species, we use the expansion (2.13) in equations of the form (2.11a) and (2.11b). Then in Eq. (2.11a), we let the functions ω range over the set of basis functions T defined in (2.15). From the definition of these B-splines, it is clear that every member of T satisfies the required continuity conditions and each member of T vanishes on ∂R_0. In Eq. (2.11b), we let the functions ω range over the set T_0 defined in 2.14. Again we see from the definition of these B-splines, that each member of T_0 when restricted to ∂R_0 satisfies the required continuity properties. For the approximation of the initial conditions we proceed in the same way in the sense that we seek an approximation for the initial conditions of the form given in Eq. (2.13). We then use this expression in (2.11c) and (2.11d) with ω ranging over the set T in (2.11c) and ω ranging over the set T_0 in (2.11d). Note that with this procedure, the approximation to the initial condition is required to satisfy the essential boundary conditions (in weak form) at the initial time $t = t_0$.

Next we consider the form of the equations generated by this procedure. First we consider the left hand side of Eq. (2.11a). For each species m, the

left hand side of Eq. (2.11a) has the following form.

$$(2.16)\quad \langle [\rho C_p]_m(t,r,z,u)\,\frac{\partial u_m}{\partial t},\, A_i B_j\rangle$$

$$= \sum_{i'=1}^{N_r}\sum_{j'=1}^{N_z} \dot{U}^m_{i'j'} \iint_R [\rho C_p]_m(t,r,z,u) A_{i'}(r) B_{j'}(z) A_i(r) B_j(z)\, d\mu_r d\mu_z$$

$$\text{for } \hat{n}_r \le i \le \hat{N}_r,\ \hat{n}_z \le j \le \hat{N}_z\ .$$

The measures are defined by $d\mu_z = dz$ and

$$d\mu_r = \begin{cases} dr\ , & \text{Cartesian geometry,} \\ 2\pi r dr\ , & \text{cylindrical geometry,} \\ 4\pi r^2 dr, & \text{spherical geometry (one dimensional only).} \end{cases}$$

For each species m, the right hand side of Eq. (2.16) has the form $\tilde{A}_m(t,U)\dot{U}^m$ where $\tilde{A}_m(t,U)$ is a rectangular matrix having $(\hat{N}_r-\hat{n}_r+1)(\hat{N}_z-\hat{n}_z+1)$ rows and N_rN_z columns with

$$U^m = U^m(t) = \{U^m_{i'j'}(t):\ 1 \le i' \le N_r,\ 1 \le j' \le N_z\} \text{ with } U = \{U^m:\ 1 \le m \le M\}.$$

Here $\dot{U}^m_{i'j'} = \frac{d}{dt} U^m_{i'j'}(t)$, and the notation $\tilde{A}_m(t,U)$ indicates that the elements of $\tilde{A}_m$ may depend on t and U through the presence of the heat capacity coefficient $[\rho C_p]_m(t,r,z,u)$.

Next we consider the right hand side of Eq. (2.11a). For each species, the right hand side is the sum of several terms. The form of each of these terms will be displayed.

2.3.1 Non-Conservative Convection Terms

Recall that θ can have either one of two values in Eq. (1.5). When $\theta = 0$ we have Eq. (1.5) in substantive derivative (non-conservative) form. In this case, the following convection term is present on the right hand side of Eq. (2.11a).

$$(2.17)\qquad -\langle \vec{V}_m \cdot \nabla u_m, A_i B_j \rangle = -\iint_R \vec{V}_m \cdot \nabla u_m A_i B_j d\mu_r d\mu_z$$

$$= -\sum_{i'=1}^{N_r} \sum_{j'=1}^{N_z} U^m_{i',j'} \iint_R A_i(r) B_j(z) \vec{V}_m \cdot \nabla (A_{i'}(r) B_{j'}(z)) d\mu_r d\mu_z ,$$

$$\text{for } \hat{n}_r \le i \le \hat{N}_r,\ \hat{n}_z \le j \le \hat{N}_z .$$

Recall that $\vec{V}_m = (V^r_m, V^z_m)$ with, for example, $V^r_m = V^r_m(t,r,z,u_1(r,z,t),\ldots, u_M(r,z,t))$.

2.3.2 Divergence Term

Recall that when $\theta = 1$, Eq. (1.5) is in conservative form so that the divergence term will include both diffusion and convection terms when $\theta = 1$. From the right hand side of Eq. (2.11a), we have

$$(2.18)\qquad -\iint_R (\vec{\mathcal{D}}_m \nabla u_m - \theta u_m \vec{V}_m) \cdot \nabla (A_i B_j) d\mu_r dz =$$

$$- \sum_{i'=1}^{N_r} \sum_{j'=1}^{N_z} U^m_{i',j'} \iint_R (\vec{\mathcal{D}}_m \nabla (A_{i'} B_{j'}) - \theta A_{i'} B_{j'} \vec{V}_m) \cdot \nabla (A_i B_j) d\mu_r dz$$

$$\text{for } \hat{n}_r \le i \le \hat{N}_r,\ \hat{n}_z \le j \le \hat{N}_z ,$$

where the explicit form of the integrand is

$$(2.19)\qquad (\mathcal{D}^r_m A'_{i'} B_{j'} - \theta A_{i'} B_{j'} V^r_m) A'_i B_j + (\mathcal{D}^z_m A_{i'} B'_{j'} - \theta A_{i'} B_{j'} V^z_m) A_i B'_j .$$

Here, $A'_{i'} = \frac{d}{dr} A_{i'}(r)$, $B'_{j'} = \frac{d}{dz} B_{j'}(z)$; and, of course, $\vec{V}_m = \vec{V}_m(t,r,z,\vec{u})$.

2.3.3 Non-Essential Boundary Terms

Recall that $\partial\tilde{R}_1$ is the union of the sides where $\beta \neq 0$; thus the integral in (2.11a) over $\partial\tilde{R}_1$ is the sum of integrals over the sides where $\beta \neq 0$.

If $\beta \neq 0$ on side 1, then we have a contribution of the form

$$(2.20)\qquad \int_{S_1} B_j\left[-(-\theta)V_m^r u_m - \beta_1^{-1}(\alpha_1 h_1 u_m - \gamma_1 h_1 \rho^0)\right] S^\delta(\underline{R})dz$$

$$= \sum_{j'=1}^{N_z} U_{1j'}^m \left[\int_{\underline{Z}}^{\overline{Z}} \{-(-\theta)V_m^r - \beta_1^{-1}\alpha_1 h_1\} B_{j'}(z)B_j(z)S^\delta(\underline{R})dz\right]$$

$$+ \beta_1^{-1}\gamma_1 \int_{\underline{Z}}^{\overline{Z}} h_1 B_j(z)\rho^0(t,\underline{R},z)S^\delta(\underline{R})dz \quad \text{for } \hat{n}_z \le j \le \hat{N}_z .$$

Here we have used the fact that on side 1, we have $\vec{V}_m \cdot \vec{n} = -V_m^r$. Note also that the transfer coefficient h_1 may depend on the materials which are present on side 1, so that h_1 is a piecewise constant function on side 1. The factor $S^\delta(\underline{R})$ is defined by

$$S^\delta(\underline{R}) = \begin{cases} 1 & \text{Cartesian geometry} \\ 2\pi R, & \text{cylindrical geometry} \\ 4\pi \underline{R}^2, & \text{spherical geometry (one dimension only).} \end{cases}$$

If $\beta \neq 0$ on side 3, then we will have a contribution from this side whose form is very similar to Eq. (2.20). In fact, we just replace $U_{1j'}^m$ by $U_{N_r j'}^m$, subscript 1 by 3, $\underline{R}$ by $\overline{R}$ and $-V_m^r$ by V_m^r in the above expression.

If $\beta \neq 0$ on side 2, we have a contribution of the following form.

$$(2.21)\qquad \int_{S_2} A_i\left[-(-\theta)V_m^z u_m - \beta_2^{-1}(\alpha_2 h_2 u_m - \gamma_2 h_2 \rho^0)\right] d\mu_r$$

$$= \sum_{i'=1}^{N_r} U_{i'1}^m \left[\int_{\underline{R}}^{\overline{R}} \{-(-\theta)V_m^z - \beta_2^{-1}\alpha_2 h_2\} A_{i'}(r)A_i(r)d\mu_r\right]$$

$$+ \beta_2^{-1}\gamma_2 \int_{\underline{R}}^{\overline{R}} h_2 A_i(r)\rho^0(t,r,\underline{Z})d\mu_r \quad \text{for } \hat{n}_r \le i \le \hat{N}_r .$$

Again, if $\beta \neq 0$ on side 4 then we have a contribution from this side which has the same form as Eq. (2.21) with $U_{i'1}^m$ replaced by U_{i',N_z}^m, subscript 2 replaced by 4, $\underline{Z}$ by $\overline{Z}$, and $-V_m^z$ by V_m^z. In Eqs. (2.20) and (2.21), the convection velocity $\vec{V}_m$ is evaluated on the appropriate boundary. Thus, for example, on side 1 in Eq. (2.20), $V_m^r = V_m^r(t,\underline{R},z,\vec{u}(t,\underline{R},z))$.

The integral over $\partial\tilde{R}_2$ in Eq. (2.11a) is treated in a manner similar to the above integrals.

2.3.4 Gap Interface Term

The gap interface term has the following form.

$$(2.22)\quad -\sum_s \int_{|\hat{\Gamma}_s^g|} \left\{(A_iB_j)^- - (A_iB_j)^+\right\} h_s^g \left\{\sum_{i'=1}^{N_r}\sum_{j'=1}^{N_z} U^m_{i',j'}\left[(A_{i'}B_{j'})^- - (A_{i'}B_{j'})^+\right]\right\}$$

$$= -\sum_s \sum_{i'=1}^{N_r}\sum_{j'=1}^{N_z} U^m_{i',j'}\left\{\int_{|\hat{\Gamma}_s^g|}\left[(A_iB_j)^- - (A_iB_j)^+\right|h_s^g\left|(A_{i'}B_{j'})^- - (A_{i'}B_{j'})^+\right]\right\}$$

for $\hat{n}_r \le i \le \hat{N}_r$, $\hat{n}_z \le j \le \hat{N}_z$.

We consider vertical and horizontal gaps separately. Let $\{\sigma^g(p): 1 \le p \le NTIR\}$ denote the set of vertical gap indices, that is, $r_{\sigma^g(p)}$ is a vertical gap interface (when $\nu^r_{\sigma^g(p)} = 0$), and let $\{\tau^g(q): 1 \le q \le NTIZ\}$ denote the set of horizontal gap indices. Recall that each breakpoint r_σ has multiplicity $d_\sigma = k_r - \nu^r_\sigma$ and a set of knots $\{\xi^r_i: IL(\sigma) - d_\sigma + 1 \le i \le IL(\sigma)\}$ associated with r_σ where $IL(\sigma)$ is the index of the last knot associated with r_σ. For example, in Fig. 2 we show a set of 5 breakpoints ($\ell = 4$) with the knots distributed on the breakpoints as indicated by the symbol x above the breakpoints. In this example, $k = 4$ so that the breakpoint r_3 could be a gap interface. For this example we have:

$\xi_1 = \xi_2 = \xi_3 = \xi_4 = r_1$, $IL(1) = 4$, $d_1 = 4$, $\nu_1 = 0$,
$\xi_5 = \xi_6 = r_2$, $IL(2) = 6$, $d_2 = 2$, $\nu_2 = 2$,
$\xi_7 = \xi_8 = \xi_9 = \xi_{10} = r_3$, $IL(3) = 10$, $d_3 = 4$, $\nu_3 = 0$,
$\xi_{11} = \xi_{12} = r_4$, $IL(4) = 12$, $d_4 = 2$, $\nu_4 = 2$, $N = 12$,
$\xi_{13} = \ldots. = \xi_{16} = r_5$, $d_5 = 4$, $\nu_5 = 0$.

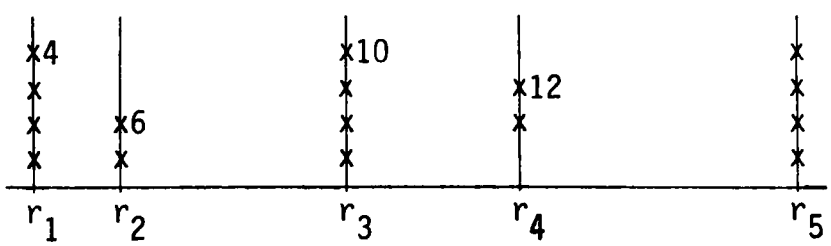


Figure 2
Sample breakpoint and knot distribution

For each vertical gap interface $r_{\sigma^g(p)}$, let $I^g(p)$ denote the index of the last knot associated with the breakpoint $r_{\sigma^g(p)-1}$, i.e., set $I^g(p) = IL(\sigma^g(p)-1)$. In the same way, for each horizontal gap interface $z_{\tau^g(q)}$, set $J^g(q) = JL(\tau^g(q)-1)$.

Consider a vertical gap with index $\sigma^g(p)$, then at this gap

$$(A_iB_j)^- - (A_iB_j)^+ = \left[A_i(r^-_{\sigma^g(p)}) - A_i(r^+_{\sigma^g(p)})\right]B_j(z) \equiv 0$$

unless $i = I^g(p)$ or $i = I^g(p)+1$, in which case

$$A_i(r^-_{\sigma^g(p)}) - A_i(r^+_{\sigma^g(p)}) = \begin{cases} 1 & \text{if } i = I^g(p), \\ -1 & \text{if } i = I^g(p)+1. \end{cases}$$

Consider the contribution to Eq. (2.11a) from the vertical gaps. With $i_0 = I^g(p)$, we have

$$\begin{aligned} (2.23a)\quad & -\sum_{p=1}^{NTIR}\sum_{i'=1}^{N_r}\sum_{j'=1}^{N_z} U^m_{i'j'}\int_{\underline{Z}}^{\overline{Z}}\left[A_i(r^-_{\sigma^g(p)}) - A_i(r^+_{\sigma^g(p)})\right]B_j(z)h^{Vg}_p \\ & \cdot\left[A_{i'}(r^-_{\sigma^g(p)}) - A_{i'}(r^+_{\sigma^g(p)})\right]B_{j'}(z)S^\delta(r_{\sigma^g(p)})dz \\ & = \begin{cases} -\sum_{p=1}^{NTIR}\sum_{j'=1}^{N_z}(U^m_{i_0j'} - U^m_{i_0+1,j'})S^\delta(r_{\sigma^g(p)})\int_{\underline{Z}}^{\overline{Z}} h^{Vg}_p B_jB_{j'}dz & \text{if } i = i_0, \\ -\sum_{p=1}^{NTIR}\sum_{j'=1}^{N_z}(-U^m_{i_0j'} + U^m_{i_0+1,j})S^\delta(r^g(p))\int_{\underline{Z}}^{\overline{Z}} h^{Vg}_p B_jB_{j'}dz & \text{if } i = i_0+1 \end{cases} \end{aligned}$$

for $\hat{n}_z \le j \le \hat{N}_z$.

In the same way, we find that horizontal gaps will contribute terms of the following form.

$$(2.23b)\quad \begin{cases} -\sum_{q=1}^{NTIZ}\sum_{i'=1}^{N_r}(U^m_{i',j_0}-U_{i',j_0+1})\int_{\underline{R}}^{\overline{R}} h_q^{Hg}A_iA_{i'}d\mu_r & \text{if } j=j_0 \\ -\sum_{q=1}^{NTIZ}\sum_{i'=1}^{N_r}(-U_{i',j_0}+U_{i',j_0+1})\int_{\underline{R}}^{\overline{R}} h_q^{Hg}A_iA_{i'}d\mu_r & \text{if } j=j_0+1 \end{cases}$$

for $\hat{n}_r \le i \le \hat{N}_r$.

2.3.5 Distributed Source Term

Recall from Eq. (1.5) that

$$(2.24)\quad F_m(t,r,z,\vec{u},\nabla\vec{u}) = \sum_{m'=1}^{M} C_{mm'}u_{m'} + \sum_{m'=1}^{M}\sum_{m''=1}^{M} C_{mm'm''}u_{m'}u_{m''} + f_m(t,r,z,\vec{u},\nabla\vec{u})$$

where

$$f_m(t,r,z,\vec{u},\nabla\vec{u}) = f_m(t,r,z,u_1(t,r,z),\ldots,u_m(t,r,z),$$
$$\frac{\partial u_1}{\partial r}(t,r,z),\ \frac{\partial u_1}{\partial z}(t,r,z),\ldots,\ \frac{\partial u_m}{\partial r}(t,r,z),\ \frac{\partial u_m}{\partial z}(t,r,z)) .$$

Thus

$$(2.25)\quad \langle F_m, A_iB_j\rangle = \iint_R F_m(t,r,z,\vec{u},\nabla\vec{u})A_iB_j d\mu_r dz \quad \text{for } \hat{n}_r \le i \le \hat{N}_r,\ \hat{n}_z \le j \le \hat{N}_z$$

We have now accounted for each term which contributes to the right hand side of Eq. (2.11a). For each species m, the right hand side of Eq. (2.11a) has a total of $(\hat{N}_r-\hat{n}_r+1)(\hat{N}_z-\hat{n}_z+1)$ components. If we denote by $\tilde{G}^m$ the vector with these components, then Eq. (2.11a) can be written as a system of $(\hat{N}_r-\hat{n}_r+1)(\hat{N}_z-\hat{n}_z+1)$ differential equations

$$(2.26)\qquad \tilde{A}_m(U)\dot{U}^m = \tilde{G}^m$$

for the N_rN_z functions $U^m = \{U^m_{i',j'}(t): 1 \le i' \le N_r,\ 1 \le j' \le N_z\}$.

2.3.6 Essential Boundary Condition Terms

This system is augmented by the weak form of the essential boundary

conditions (2.11b) which generates an additional system of algebraic equations as ω ranges over the set T_0. Thus if $s_1 = 1$, then the set $\{A_1(r)B_j(z): \hat{n}_z \le j \le \hat{N}_z\}$ is part of T_0 and Eq. (2.11b) will provide the following equations.

$$(2.27a) \quad \sum_{j'=1}^{N_z} U^m_{1j'} S^\delta(\underline{R}) \int_{\underline{Z}}^{\overline{Z}} B_{j'}B_j dz = \frac{\gamma_1}{\alpha_1} S^\delta(\underline{R}) \int_{\underline{Z}}^{\overline{Z}} B_j \rho^0(t,\underline{R},z)dz \quad \text{for } \hat{n}_z \le j \le \hat{N}_z .$$

If $s_2 = 1$ then the set $\{A_iB_j: 1 \le i \le N_r\}$ is part of T_0; so that Eq. (2.11b) will provide the following equations.

$$(2.27b) \quad \sum_{i'=1}^{N_r} U^m_{i',1} \int_{\underline{R}}^{\overline{R}} A_{i'}A_i d\mu_r = \frac{\gamma_2}{\alpha_2} \int_{\underline{R}}^{\overline{R}} A_i \rho^0(t,r,\underline{Z})d\mu_r \quad \text{for } 1 \le i \le N_r .$$

If $s_3 = 1$, we obtain the set

$$(2.27c) \quad \sum_{j'=1}^{N_z} U^m_{N_r,j'} S^\delta(\overline{R}) \int_{\underline{Z}}^{\overline{Z}} B_{j'}B_j dz = \frac{\gamma_3}{\alpha_3} S^\delta(\overline{R}) \int_{\underline{Z}}^{\overline{Z}} \rho^0(t,\overline{R},z)B_j dz \quad \text{for } \hat{n}_z \le j \le \hat{N}_z .$$

If $s_4 = 1$, we obtain the set

$$(2.27d) \quad \sum_{i'=1}^{N_r} U^m_{i',N_z} \int_{\underline{R}}^{\overline{R}} A_{i'}A_i d\mu_r = \frac{\gamma_4}{\alpha_4} \int_{\underline{R}}^{\overline{R}} A_i \rho^0(t,r,\overline{Z})d\mu_r \quad \text{for } 1 \le i \le N_r .$$

Equations 2.27 provide an additional system of $(s_1+s_3)(\hat{N}_z-\hat{n}_z+1) + (s_2+s_4)N_r$ algebraic equations which when considered together with the system of differential equations in (2.26) provides us with a mixed system of algebraic and differential equations whose total number is N_rN_z.

The algebraic equations (2.27) can be avoided if in the essential boundary condition $u = \frac{\gamma}{\alpha}\rho^0$, we take the partial derivative with respect to time of both sides of this equation. We then have

$$\dot{u} = \gamma/\alpha \, \dot{\rho}^0,$$

and the weak form

$$\int_{\partial R_0} \omega \dot{u} = \int_{\partial R_0} \frac{\gamma}{\alpha} \dot{\rho}^0 \omega \ . \tag{2.28}$$

From this expression we then obtain a system of differential equations which have the same form as the system (2.27) with $U^m_{i',j'}$ replaced by $\dot{U}^m_{i',j'}$ and ρ^0 replaced by $\dot{\rho}^0 = \partial\rho^0/\partial t$. When this differential system is combined with the system (2.26) we obtain a system of $N_r N_z$ differential equations in $N_r N_z$ unknown functions for each species. This system can be written in the form

$$A_m(U)\dot{U}^m = G^m, \quad m=1,\ldots,M \tag{2.29}$$

where $A_m(U)$ is now a square matrix.

The mixed differential and algebraic system will be called the mixed or algebraic boundary condition version. The differential system (2.29) will be called the differential boundary condition version. Both versions have been implemented in the same program, and either can be selected on input.

2.3.7 Initial Conditions

The initial conditions are generated from the weak form given by Eqs. (2.11c,d) as follows. Given an initial distribution $u^0_m(r,z)$, we seek the projection of this function into the space spanned by the sets T and T_0. That is, we seek

$$\hat{u}^0_m(r,z) = \sum_{i'=1}^{N_r} \sum_{j'=1}^{N_z} \hat{U}^m_{i',j'} A_{i'} B_{j'} \ , \tag{2.30}$$

and determine the coefficients $\hat{U}^m_{i',j'}$ by the following equations.

$$\langle \hat{u}^0, A_i B_j \rangle = \sum_{i'=1}^{N_r} \sum_{j'=1}^{N_z} \hat{U}^m_{i',j'} \langle A_{i'} B_{j'}, A_i B_j \rangle = \langle u^0, A_i B_j \rangle \tag{2.31}$$

$$\text{for } \hat{n}_r \le i \le \hat{N}_r, \ \hat{n}_z \le j \le \hat{N}_z$$

and if $s_1 = 1$ (side 1 is essential, $\beta = 0$)

$$(2.32)\qquad \sum_{j'=1}^{N_z} \hat{U}^m_{1j'} S^\delta(\underline{R}) \int_{\underline{Z}}^{\overline{Z}} B_j B_{j'} dz = S^\delta(\underline{R}) \int_{\underline{Z}}^{\overline{Z}} \frac{\gamma_1}{\alpha_1} B_j \rho^0(t_0,\underline{R},z)dz \quad \text{for } \hat{n}_z \le j \le \hat{N}_z ,$$

if $s_2 = 1$ ($\beta = 0$ on side 2)

$$(2.33)\qquad \sum_{i'=1}^{N_r} \hat{U}^m_{i'1} \int_{\underline{R}}^{\overline{R}} A_i A_{i'} d\mu_r = \int_{\underline{R}}^{\overline{R}} \frac{\gamma_2}{\alpha_2} A_i \rho^0(t_0,r,\underline{Z})d\mu_r \quad \text{for } 1 \le i \le N_r ,$$

with similar expressions if $s_3 = 1$ and if $s_4 = 1$. Note that in determining $\{\hat{U}^m_{ij}\}$ we have imposed the initial values of the essential boundary conditions (i.e. $\rho^0(t_0)$ on ∂R_0) even though the initial distribution u^0 may or may not satisfy these conditions. Using $\rho^0(t_0)$ on ∂R_0 means that we have selected one of many possible projections of u^0 into the space spanned by the sets T and T_0.

2.4 Evaluation of Integrals

In the case of algebraic constraints on the essential boundaries, the Galerkin procedure leads to a mixed system of ordinary differential equations and algebraic equations. In the case of differential constraints, the Galerkin procedure leads to a system of ordinary differential equations of the form given in Eq. (2.29). In either case, a variety of integrals have to be evaluated. The general procedure for the evaluation is the same for all of these integrals and can be illustrated by considering the integrals appearing in the coefficient matrix of Eq. (2.29) which arise from Eq. (2.16). From this equation, we see that the following integrals have to be computed.

$$(2.34)\quad I^A(i,j;i',j') = \iint_R [\rho C_p](t,r,z,u) A_{i'}(r) A_i(r) B_{j'}(z) B_j(z) d\mu_r dz$$
$$\text{for } \hat{n}_r \le i \le \hat{N}_r,\ \hat{n}_z \le j \le \hat{N}_z, \text{ and } 1 \le i' \le N_r,\ 1 \le j' \le N_z.$$

First we observe that since $A_i(r)A_{i'}(r) \equiv 0$ for $|i-i'| > k_r$ and $B_j(z)B_{j'}(z) \equiv 0$ for $|j-j'| > k_z$, we have $I^A(i,j;i',j') = 0$ when $|i-i'| > k_r$ or $|j-j'| > k_z$. (Recall from (2.12 iv) that $A_i(r)$ has its support in the interval $[\xi^r_i, \xi^r_{i+k_r}]$.) The computation of the integrals is accomplished by

accumulating the integrals over each mesh subrectangle. Thus

$$(2.35)\qquad I^A(i,j;i',j') = \sum_{\sigma=1}^{\ell_r}\sum_{\tau=1}^{\ell_z} I^A_{\sigma,\tau}(i,j;i',j') \ ,$$

where

$$(2.36)\qquad I^A_{\sigma,\tau}(i,j;i',j') = \int_{z_\tau}^{z_{\tau+1}}\int_{r_\sigma}^{r_{\sigma+1}} [\rho C_p](t,r,z,u)A_i(r)A_{i'}(r)B_j(z)B_{j'}(z)d\mu_r d\mu_z .$$

If $IL(\sigma)$ is the index of the last knot associated with r_σ then if $r \in [r_\sigma, r_{\sigma+1}]$ we have $A_{i''}(r) = 0$ for $i'' \notin [IL(\sigma)-k_r+1, IL(\sigma)]$. Similarly, let $JL(\tau)$ denote the index of the last knot associated with z_τ, then if $z \in [z_\tau, z_{\tau+1}]$, we have $B_{j''}(z) = 0$ for $j'' \notin [JL(\tau)-k_z+1, JL(\tau)]$. Considering the integrals in (2.36), we see then that when $r \in [r_\sigma, r_{\sigma+1}]$, $A_i(r)A_{i'}(r) \neq 0$ for $\hat{i}_\ell \leq i \leq \hat{i}_u$ and $i_\ell \leq i' \leq i_u$ where $i_\ell = IL(\sigma)-k_r+1$, $i_u = IL(\sigma)$, $\hat{i}_\ell = \text{Max}(\hat{n}_r, i_\ell)$, and $\hat{i}_u = \text{Min}(\hat{N}_r, i_u)$. In the same way, we find that for $z \in [z_\tau, z_{\tau+1}]$, $B_j(z)B_{j'}(z) \neq 0$ for $\hat{j}_\ell \leq j \leq \hat{j}_u$ and $j_\ell \leq j' \leq j_u$ where j_ℓ, j_u, etc. are defined as above with $JL(\tau)$ and k_z in place of $IL(\sigma)$ and k_r. Thus for each σ, $1 \leq \sigma \leq \ell_r$ and τ, $1 \leq \tau \leq \ell_z$, an integral of the type 2.36 must be evaluated for

$$i_\ell \leq i' \leq i_u, \quad \hat{i}_\ell \leq i \leq \hat{i}_u, \quad j_\ell \leq j' \leq j_u, \quad \text{and} \quad \hat{j}_\ell \leq j \leq \hat{j}_u \ .$$

Since each integral extends over a rectangular region, a product formula is a natural choice for a numerical quadrature scheme. This program uses a product Gauss-Legendre quadrature scheme with NQR points in each interval $[r_\sigma, r_{\sigma+1}]$ and NQZ points in each interval $[z_\tau, z_{\tau+1}]$. The values of NQR and NQZ can be selected by the user with the restriction that NQR $\geq k_r-1$ and NQZ $\geq k_z-1$. However, NQR = k_r and NQZ = k_z are reasonable choices for these values in the sense that for this choice the error due to the use of quadrature formulas is much less than the error due to the Galerkin approximation. The choice NQR = k_r-1 and NQZ = k_z-1, when it works, appear to be optimal in the sense that the quadrature error is not greater than the Galerkin approximations error. This would be in agreement with the theory for elliptic problems as discussed by Strang in [4]. However, there are problems for which this choice does not work. See, for example, the sample problem in section 7.6; in this case one must use the default values NQR = k_r and NQZ = k_z (cf. § 5.2).

3. PROBLEM DESCRIPTION

In this section, we describe the class of problems which can be solved by DISPL, as well as its capabilities and limitations. We will also describe input and output of the code. In short then, this section and the remainder of this report constitutes a user's guide.

3.1 Domain

$$R = \{(r,z): \text{RLOW} \le r \le \text{RUP}, \ \text{ZLOW} \le z \le \text{ZUP}\}$$

is a rectangular domain with sides parallel to the coordinate axes.

3.2 Geometry

$$\text{DELTA} = \begin{cases} 0, & \text{Cartesian } (x,y), \\ 1, & \text{cylindrical } (r,z), \\ 2, & \text{spherical (one-dimensional only).} \end{cases}$$

3.3 Interfaces

The domain R may be composed of subrectangles such that each subrectangle has its own material properties.

NTIR ... the number of vertical interfaces.

NTIZ ... the number of horizontal interfaces.

NTIR = 0 (NTIZ = 0) means that there are no vertical (horizontal) interfaces.

RIF(I), $1 \le I \le$ NTIR ... the position of the I-th vertical interface.

ZIF(J), $1 \le J \le$ NTIZ ... the position of the J-th horizontal interface.

3.4 Additional Mesh Points

In addition to the interfaces, the domain R can be subdivided further by additional mesh points.

NMR ... the total number of additional vertical mesh points in the domain R not including the end points.

NMZ ... the total number of additional horizontal mesh points in the domain R not including the end points.

RMESH(I), $1 \le I \le$ NMR ... the position of the I-th vertical additional mesh point. Here RLOW < RMESH(I) < RUP.

ZMESH(J), $1 \le J \le$ NMZ ... the position of the J-th horizontal additional mesh point. Here ZLOW < ZMESH(J) < ZUP.

The program merges the additional mesh points with the interface points to form a mesh over which the B-splines are defined.

3.5 Partial Differential Equations

The basic equation considered by this program can be viewed as a general form of the equation of continuity of a multicomponent fluid under the assumption that the total mass density is constant. This particular physical model provides convenient terminology for describing the equations. From a mathematical standpoint, the equation considered by this program is a system of nonlinear parabolic equations in two spatial variables.

NSPEC ... denotes the number of species (total number of parabolic partial differential equations).

$u_m = u_m(t,r,z)$ ... denotes the concentration of the m-th species at the point (t,r,z) (dependent variables for the m-th equation).

$\vec{u} = (u_1,u_2,\ldots,u_m)^T$ ... denotes the vector of concentrations.

The system of equations has the following form.

$$(3.1)\quad [\rho C_p]_m(t,r,z,\vec{u})\frac{\partial u_m}{\partial t} + \theta\nabla\cdot(\vec{V}_m(t,r,z,\vec{u})u_m) + (1-\theta)\vec{V}_m(t,r,z,\vec{u})\cdot\nabla u_m$$
$$= \nabla\cdot(\vec{D}_m(t,r,z,\vec{u},\overrightarrow{\nabla u})\, u_m) + \sum_{m'=1}^{NSPEC} c_{mm'}u_{m'}$$
$$+ \sum_{m'=1}^{NSPEC}\sum_{m''=1}^{NSPEC} c_{mm'm''}\, u_{m'}\, u_{m''} + f_m(t,r,z,\vec{u},\overrightarrow{\nabla u})$$
$$\text{for } 1 \le m \le \text{NSPEC}\ .$$

When $[\rho C_p]_m(t,r,z,\vec{u}) \equiv 1$ for any m, the program precomputes the integrals I^A appearing in (2.34) which can result in a substantial reduction in the execution time. In Namelist GRID, the flag IREVLA(m)=T, implies that $[\rho C_p]_m$ is not identically 1 or 0. IREVLA(m)=F, implies that $[\rho C_p]_m$ is either

identically 1 or 0. The flag IRHØ(m)=T, implies that $[\rho C_p]_m \equiv 0$ while IRHØ(m)=F, implies $[\rho C_p]_m \neq 0$. The parameter θ can have either of the values 0 or 1. When $\theta = 1$, we have the conservative form, and when $\theta = 0$, we have the non-conservative form for the equations. The conservative form is used by the program by setting the logical indicator CØNSRV=T, in the input namelist GRID.

The following user-supplied subroutines provide the coefficient functions appearing in Eq. (3.1).

RHOCP - supplies the heat capacity coefficient $[\rho C_p]_m(t,r,z,\vec{u})$.

VEL - supplies the convection velocity vector coefficients $\vec{V}_m(t,r,z,\vec{u})$.

DIFUSE - supplies the diffusion vector coefficient $\vec{\mathcal{D}}_m(t,r,z,\vec{u},\overrightarrow{\nabla u})$.

EXTSRC - supplies the distributed source $f_m(t,r,z,\vec{u},\overrightarrow{\nabla u})$.

In addition to the indicated dependence of these functions are the arguments t, r, z, $\vec{u}$, and $\overrightarrow{\nabla u}$, each of these functions can also depend on the material index present at the point (r,z). This dependence on the material index can simplify the task of writing these subroutines when the coefficients depend on the materials which are present.

3.6 Interface Conditions

If interfaces are present in the domain R, then interface conditions must be applied across each interface. Recall that NTIR is the number of vertical interfaces and NTIZ is the number of horizontal interfaces. (If NTIR (NTIZ) is zero, then there are no vertical (horizontal) interfaces in the domain R.) As far as this program is concerned, an interface must extend from one external boundary to the opposite external boundary of R. Let Γ denote an interface, then this program allows for one of two possible interface conditions to be imposed on Γ.

3.6.1 Continuity Condition (continuity of density and flux)

$$(3.2)\qquad \begin{cases} \text{(i)} \quad u_m\big|_{\Gamma^-} = u_m\big|_{\Gamma^+} \\ \text{(ii)} \quad \vec{\mathcal{D}}_m \dfrac{u_m}{\partial |n|}\Big|_{\Gamma^-} = \vec{\mathcal{D}}_m \dfrac{\partial u_m}{\partial |n|}\Big|_{\Gamma^+}, \quad \text{for} \quad 1 \le m \le \text{NSPEC}. \end{cases}$$

Here $\vec{\mathcal{V}}_m = \vec{\mathcal{V}}_m(t,r,z,\vec{u}, u)$ and $\vec{\mathcal{V}}_m \frac{\partial u_m}{\partial |n|} = \mathcal{V}_m^r \frac{\partial u_m}{\partial r}$ if Γ is vertical; while $\vec{\mathcal{V}}_m \frac{\partial u_m}{\partial |n|} = \mathcal{V}_m^z \frac{\partial u_m}{\partial z}$ if Γ is horizontal. In addition, the symbol $|_\Gamma+$ indicates a limiting value taken from the right (above) if Γ is vertical (horizontal) with the corresponding meaning for $|_\Gamma-$.

3.6.2 Gap Condition (Discontinuity in the density and continuity in the flux.)

$$(3.3)\quad \begin{cases} \text{(i)} & -\vec{\mathcal{V}}_m \dfrac{\partial u_m}{\partial |n|}\Big|_{\Gamma-} = h^g\left\{u_m\Big|_{\Gamma-} - u_m\Big|_{\Gamma+}\right\} \\ \text{(ii)} & \vec{\mathcal{V}}_m \dfrac{\partial u_m}{\partial |n|}\Big|_{\Gamma-} = \vec{\mathcal{V}}_m \dfrac{\partial u_m}{\partial |n|}\Big|_{\Gamma+} \end{cases}$$

Let NIGAP denote the set of vertical gap interfaces. The I-th vertical gap will intersect a set of NTIZ horizontal interfaces. This horizontal set of interfaces will subdivide the I-th vertical gap into a set of 1 + NTIZ subintervals. The vertical gap coefficients h^{Vg} can then depend on the following parameters.

$$(3.4i)\quad h^{Vg} = \text{HVGAP(m,I,J)} \quad \text{for} \quad 1 \le m \le \text{NSPEC}, \quad 1 \le I \le \text{NIGAP}, \text{ and } 1 \le J \le 1+\text{NTIZ}.$$

In the same way, if NJGAP denotes the number of horizontal gaps, then

$$(3.4ii)\quad h^{Hg} = \text{HHGAP(m,J,I)} \quad \text{for} \quad 1 \le m \le \text{NSPEC}, \quad 1 \le J \le \text{NJGAP}, \text{ and } 1 \le I \le 1+\text{NTIR}.$$

The gap coefficients HVGAP(m,I,J) and HHGAP(m,J,I) are supplied by the user in the input namelist DATA.

With regard to interfaces, this program is restricted by the following requirements.

(i) All interfaces must be parallel to the coordinate axes.

(R3.1) (ii) Each interface must extend from one external boundary to the opposite external boundary.

(iii) For a given species and a given interface, the same type of interface condition must be applied everywhere on the interface.

In order to illustrate the last two restrictions, consider the following domain with three different materials labeled I, II, and III as shown in Fig. 3.

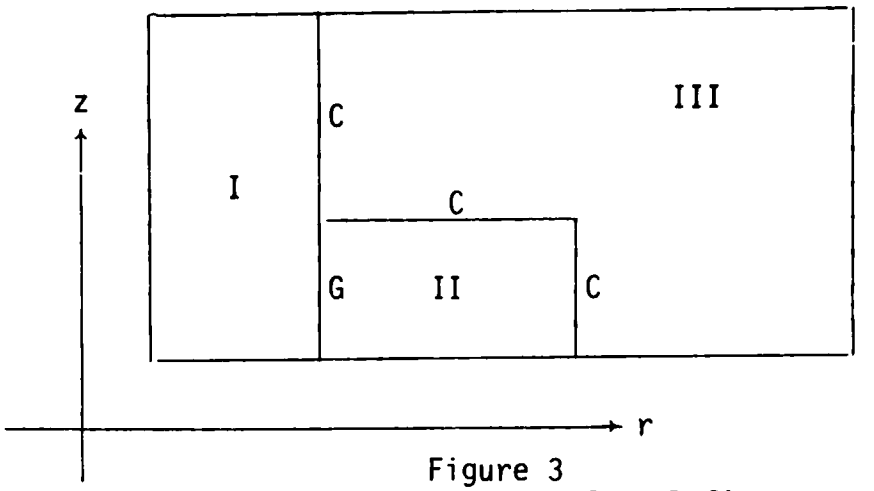


Figure 3
Interface Conditions before Refinement

Suppose that for a given species, we have a gap condition between materials I and II, and continuity conditions between materials I and III as well as material II and III. As it stands this configuration cannot be handled by this program. However, an approximate problem can be handled by the program. To this end we first extend the interfaces so that they extend from one exterior boundary to the opposite side. This gives a set of NTIR=2 vertical interfaces and a set of NTIZ=1 horizontal interfaces as shown in Fig. 4.

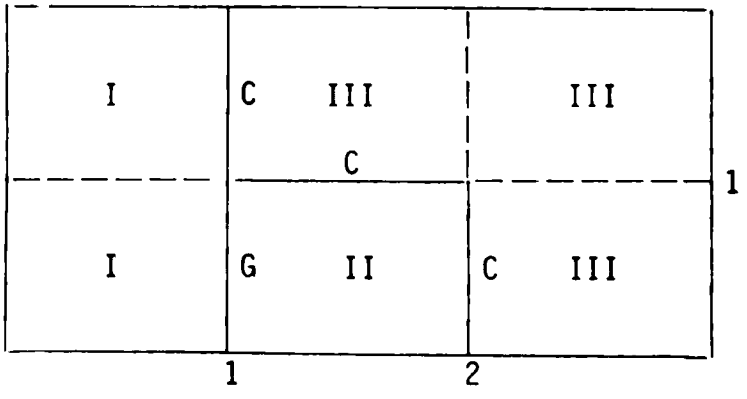


Figure 4
Interface Conditions after Refinement

Consider the 1st vertical interface which is divided into two subintervals. The lower subinterval separates materials I and II and across this interface we have a gap condition. But then the program requires a gap condition to be applied across the upper subinterval which separates materials I and III. However, the original problem required a continuity condition across this upper subinterval. In order to approximate the original problem, we observe that the continuity condition is a limiting case of a gap condition as the gap coefficient increases in magnitude. Thus, in order to approximate the original problem, we would impose a gap condition across the first vertical interface. There are two gap coefficients associated with this interface $h^{Vg}(m,1,J)$, J=1,2, with $h^{Vg}(m,1,1)$ the given coefficient for materials I and II, and $h^{Vg}(m,1,2)$ an arbitrary but large number. If $h^{Vg}(m,1,2)$ is sufficiently large, the original interface condition across the first vertical interface will be approximated as closely as desired. Extending the horizontal interface implies that we have introduced an interface in material I, for example, where in the original problem there was none. Interfaces of this type would cause concern only if very accurate values of the concentration were required in the vicinity of corners.

3.7 Boundary Conditions

For each species m, a boundary condition may be specified on each of the four sides of the domain R. The boundary conditions have the following form.

$$\alpha h u_m + \beta \vec{\mathcal{D}}_m \nabla u_m \cdot \vec{\hat{n}} = \gamma h \rho_m^0 \tag{3.5}$$

where $\vec{\hat{n}}$ denotes the unit exterior normal on R, and

$$\left.\begin{aligned} \alpha &= \text{ALPHA(m,s)} \\ \beta &= \text{BETA(m,s)} \\ \gamma &= \text{GAMMA(m,s)} \end{aligned}\right\} \quad 1 \le m \le \text{NSPEC}, \quad 1 \le s \le 4 ,$$

are specified in namelist DATA for each species m and each side index s. The sides are numbered counterclockwise starting with the left hand side. The mass transfer coefficients are indexed as follows.

$$h = \begin{cases} \text{HU1(J,m) on side 1} \\ \text{HU3(J,m) on side 3} \end{cases} \quad 1 \le J \le 1 + \text{NTIZ}, \quad 1 \le m \le \text{NSPEC}.$$

$$h = \begin{cases} \text{HU2(I,m) on side 2} \\ \text{HU4(I,m) on side 4} \end{cases} \quad 1 \le I \le 1 + \text{NTIR}, \quad 1 \le m \le \text{NSPEC}.$$

These coefficients are specified in the input namelist DATA.

The function $\rho^0(t,m,s,x)$ is specified in the user-supplied subroutine BRHO when the algebraic version is used (ALGBCS=T,). The time derivative of ρ_m^0 is supplied in the user-supplied subroutine BRHODT when the differential version is used (ALGBCS=F,). The value of ρ_m^0 is allowed to depend on the normal component $\vec{V}_m \cdot \vec{n}$ of the convection velocity. In addition, the function ρ^0 can depend on $\vec{u}$ and the derivatives $\overrightarrow{\nabla u} \cdot |\vec{n}|$ evaluated at x on side s. That is,

$$\rho^0 = \rho^0(t,m,s,x,u_1(x),\ldots,u_M(x), \overrightarrow{\nabla u}_1 \cdot |\vec{n}|(x),\ldots,\overrightarrow{\nabla u}_n \cdot |\vec{n}|(x)) \ .$$

Note that if $g(t,m,s,x,\vec{u}(x),\overrightarrow{\nabla u}\cdot\vec{u}(x)) = 0$ is a given nonlinear boundary condition, then we can achieve the form of equation (3.5) by setting $\rho_m^0 = g(t,m,s,x,\vec{u}(x),\overrightarrow{\nabla u}\cdot\vec{u}(x)) + \alpha u_m + \beta(\mathcal{D}_m \overrightarrow{\nabla u}_m)\cdot\vec{n}(x)$ with $h \equiv 1$ and $\gamma \equiv 1$. It should be emphasized that boundary conditions must be put in the form given by equation (3.5).

In order to avoid any confusion on signs, we write equation (3.5) explicitly for each side.

$$\alpha_1 h_1 u_m - \beta_1 \left(\mathcal{D}_m^r \frac{\partial u_m}{r}\right) = \gamma_1 h_1 \rho_{1,m}^0 \ ,$$

$$\alpha_2 h_2 u_m - \beta_2 \left(\mathcal{D}_m^z \frac{\partial u_m}{z}\right) = \gamma_2 h_2 \rho_{2,m}^0 \ ,$$

$$\alpha_3 h_3 u_m + \beta_3 \left(\mathcal{D}_m^r \frac{\partial u_m}{r}\right) = \gamma_3 h_3 \rho_{3,m}^0 \ ,$$

$$\alpha_4 h_4 u_m + \beta_4 \left(\mathcal{D}_m^z \frac{\partial u_m}{\partial z}\right) = \gamma_4 h_4 \rho_{4,m}^0 \ .$$

For each species m, the program requires a set of four indicators, provided in the input namelist DATA, with the following meaning.

$$NSJ(m) = \begin{cases} 1 \text{ if side J has essential boundary conditions} \\ 0 \text{ if side J does not have essential boundary conditions} \\ -1 \text{ if side J does not have a boundary condition} \end{cases}$$

for J = 1,2,3,4; and m = 1,NSPEC.

3.8 Initial Conditions

This program allows for two possible types of initial conditions.

A. An arbitrary initial distribution $\{u_m^0(r,z): 1 \leq m \leq NSPEC\}$ can be specified in a user-supplied subroutine INDATA. The program will then project this data into the approximating subspace in order to provide the initial data for solving the system of ordinary differential equations. The use of this option is indicated by setting INITSW=T, in the input namelist GRID.

B. The second type of initial condition is for the program to start from some particular steady-state or equilibrium distribution $\{\tilde{u}_m^0(r,z): 1 \leq m \leq NSPEC\}$. The program will first compute an approximation to this steady-state distribution in the approximating subspace of B-splines by means of the control subroutine STEADY. The program can then use this steady-state solution as the initial data for a transient calculation which is done under the control of subroutine TIMEX. Recall that if all the coefficients (convection velocity, diffusivity, distributed source, and external boundary functions) are independent of time, then the steady solution exists and can be found by integrating the differential equations out in time until the solutions are independent of time. The program uses this approach to find a steady-state solution. A steady-state calculation is indicated by setting STEDSW=T, in the input namelist GRID. If this calculation is to be followed by a transient calculation which uses the steady-state solution as its initial data, then this is indicated by setting TRANSW=T, in addition to STEDSW=T. When the program performs a steady-state calculation, it will require an initial estimate for the steady-state solution. There are two options available for providing this initial estimate. First, if the user has an initial estimate, he can provide this estimate in subroutine INDATA and signal the program to use this estimate by setting INITSW=T, in the input

namelist GRID. If the user does not wish to provide an initial estimate for the steady-state calculation, the program will generate an initial estimate with the control program GUESS1. This option can be invoked by setting GUESSW=T, in the input namelist GRID.

The four switches TRANSW, STEDSW, INITSW, and GUESSW control the nature of the calculation as well as the nature of the initial conditions. We illustrate this with some examples.

I. GUESSW=F, INITSW=T, STEDSW=T, TRANSW=T. This indicates that a transient calculation is to be done with the initial data provided by the result of a steady-state calculation. Moreover, the initial estimate for the steady-state calculation is provided by the user in the user subroutine INDATA.

II. GUESSW=T, INITSW=F, STEDSW=T, TRANSW=T. This is the same calculation as in I. except that the user does not supply the initial estimate for the steady-state calculation.

III. GUESSW=F, INITSW=T, STEDSW=F, TRANSW=T. This is a transient calculation with the initial data provided by the user in the user subroutine INDATA.

All four switches have default values T; thus each switch must be explicitly set to F if that calculation is not desired.

The physical conditions which initiate a transient can be provided in the appropriate user-supplied subroutines or in the input namelist DATA whichever is applicable for initiating the transient. Note that two different sets of input data can be provided for namelist DATA. The first set is used in a steady-state calculation and the second set is used in the transient calculation.

4. DESCRIPTION OF USER-SUPPLIED SUBROUTINES

This program requires nine user-supplied subroutines each written in FORTRAN. Every one of these routines must be present in at least dummy form.

4.1 Subroutine DIFUSE(IMATL,KSPEC,NSPEC,T,RR,ZZ,SPDEN,SPDENR,SPDENZ,DIFUR, DIFUZ,TO)

Variable Names and Meanings:

KSPEC....Species index.

NSPEC....Total number of species.

T.....Current value of the time.

RR.....Value of the abscissa.

ZZ.....Value of the ordinate.

IMATL....Value of the material index at the position (RR,ZZ).

SPDEN....An array SPDEN(K), K=1,NSPEC for which SPDEN(K) = u(K,T,RR,ZZ) is the concentration of the K-th species at time T and position (RR,ZZ).

SPDENR...An array SPDENR(K), K=1,NSPEC for which SPDENR(K) = $\frac{\partial u}{\partial r}$(K,T,RR,ZZ).

SPDENZ...An array SPDENZ(K), K=1,NSPEC for which SPDENZ(K) = $\frac{\partial u}{\partial z}$(K,T,RR,ZZ).

DIFUR....Output value of the r-component of the diffusion coefficient for the species with index KSPEC.

DIFUZ....Output value of the z-component of the diffusion coefficient for the species with index KSPEC.

TO.....Initial value of time at which a transient calculation starts. This value can be used to distinguish whether a steady-state (T < TO) calculation is in progress or whether a transient (T $\geq$ TO) calculation is in progress.

Given T, RR, ZZ, IMATL, and {SPDEN(K),SPDENR(K),SPDENZ(K): K=1,NSPEC}, this routine returns the two components of the diffusion coefficient for the species with index KSPEC.

When a steady-state calculation is in progress we have T < TO and this routine must then return a diffusion coefficient which is independent of time.

Moreover, if the user does not provide an initial estimate (INITSW=F in namelist GRID), then the GUESS1 option (GUESSW=T in namelist GRID) must be used to provide an initial estimate for the steady-state calculation. In the course of the GUESS1 calculation, the program requires that this subroutine provides an initial estimate for the diffusion coefficient. Since the concentrations are not known, this initial estimate cannot depend on SPDEN, SPDENR, SPDENZ, or T. The flag IPHASE=-2 is used to indicate when this initial estimate is to be provided for a GUESS1 calculation. This indicator is transmitted through the CØMMØN block BNDCØM which can appear in this subroutine.

The general form of DIFUSE could be as follows.

```
      SUBRØUTINE DIFUSE(IMATL,KSPEC,NSPEC,T,RR,ZZ,SPDEN,SPDENR,SPDENZ,
     #     DIFUR,DIFUZ,TO)
      INTEGER IMATL,KSPEC,NSPEC
      DØUBLE PRECISIØN T,RR,ZZ,SPDEN(NSPEC),SPDENR(NSPEC),SPDENZ(NSPEC),
     #     DIFUR,DIFUZ,TO
      CØMMØN/BNDCØM/IPHASE,NS1(MAXSP),NS2(MAXSP),NS3(MAXSP),NS4(MAXSP)
      INTEGER IPHASE,NS1,NS2,NS3,NS4
      IF (IPHASE .EQ. -2) GO TO 10
      IF (T .LT. TO) GØ TØ 5
      DIFUR = D(IMATL,KSPEC,T,RR,ZZ,SPDEN(1),...,SPDEN(NSPEC),
     #     SPDENR(1),...,SPDENZ(NSPEC))
      DIFUZ = DIFUR
      RETURN
    5 DIFUR = DH(IMATL,RR,ZZ,SPDEN(1),...,SPDENZ(NSPEC))
      DIFUZ = DIFUR
      RETURN
   10 DIFUR = DW(IMATL,RR,ZZ)
      DIFUZ = DIFUR
      RETURN
      END
```

where MAXSP in COMMON block BNDCØM is the value used in the MORTRAN macros (cf. §5.1).

Here D is a known expression for the diffusion coefficient which is to be used during a transient calculation (TRANSW=T in namelist GRID). In the same way DH is a known expression to be used during a steady-state calculation (STEDSW=T), and DW is to be used during a GUESS1 calculation (GUESSW=T).

4.2 Subroutine VEL(IMATL,KSPEC,NSPEC,T,RR,ZZ,SPDEN,VELR,VELZ,TO)

Variable Names and Meanings:

KSPEC....Species index.

NSPEC....Total number of species.

T.....Current value of the time.

RR.....Value of the abscissa.

ZZ.....Value of the ordinate.

IMATL....Value of the material index at the position (RR,ZZ).

SPDEN....The array {SPDEN(K): K=1,NSPEC} of concentrations.

VELR.....Output value of the r-component of the convection velocity.

VELZ.....Output value of the z-component of the convection velocity.

TO.....Initial value of time at which a transient calculation starts.

Given T, RR, ZZ, IMATL, and {SPDEN(K), K=1,NSPEC}, this routine returns the two components of the convection velocity. Note that in Eq. (3.1), the convection term appears on the left side of this equation; therefore, this routine must return values for VELR and VELZ consistent with these terms appearing on the left side.

All remarks concerning subroutine DIFUSE also apply to subroutine VEL. The general form of VEL would be the same as that of DIFUSE with the obvious changes.

4.3 Subroutine BRHØ(T,KSPEC,NSPEC,ISIDE,XX,VLBD,SPDEN,SPDENX,RHØV,TO)

Variable Names and Meanings:

KSPEC....Species index.

NSPEC....Total number of species.

ISIDE....Side index. This index can have any integer value from one to four. The sides of the domain are indexed counterclockwise starting with the left side.

XX.....The position coordinate of a point on the side with index ISIDE. Thus if ISIDE=1 or 3, XX=Z is the ordinate of a point

on either of these sides. If ISIDE=2 or 4, then XX=R is the abscissa of a point on either of these sides.

VLBD.....Normal component of the convection velocity at the position XX on side ISIDE for species KSPEC. That is, VLBD=V^r on sides 1 and 3, VLBD=V^z on sides 2 and 4.

SPDEN....The array {SPDEN(K), K=1,NSPEC} of concentrations.

SPDENX...The array {SPDENX(K), K=1,NSPEC} of derivatives $\frac{\partial u}{\partial |n|}(K,T,XX)$ of u on side ISIDE evaluated at time T and position XX on ISIDE. That is, $\frac{\partial u}{\partial r}$ on sides 1 and 3, $\frac{\partial u}{\partial z}$ on sides 2 and 4.

RHØV.....Output value of the boundary function on side ISIDE at position XX for species KSPEC at time T.

TO.......Initial time at which a transient calculation starts.

Given T, KSPEC, ISIDE, XX, VLBD, SPDEN, and SPDENX, this routine returns the boundary values appearing on the right side of Eq. (3.5), i.e., the user-supplied function ρ^0(T,KSPEC,ISIDE,XX). Note that since {SPDEN(K)} and {SPDENX(K)} are available, the value for ρ^0(T,KSPEC,ISIDE,XX) can depend on the values of the arrays SPDEN and SPDENX; thus nonlinear boundary conditions are allowed. In problems where convective flow is important, it may be useful to have ρ^0 depend on the sign of the normal component of the convection velocity. For example, if on side 1, we require that ρ^0(T,KSPEC,1,XX) = 0 when VLBD > 0, and on side 3, we require that ρ^0(T,KSPEC,3,XX) = 0 when VLBD < 0; then with α=0, β=γ=1, Eq. (3.5) states that there is no incoming flux on sides 1 and 3.

Recall that when a steady-state calculation is in progress, the boundary function ρ^0(T,KSPEC,ISIDE,XX) must be independent of the time T. A steady-state calculation is in progress when T < TO and a transient calculation is in progress when T $\geq$ TO. This routine is called when the algebraic version (ALGBCS=T, in namelist GRID) is used or when an initial solution estimate via the user-supplied subroutine INDATA (INITSW=T, in Namelist GRID) is used.

The general form of BRHØ could be as follows.

```
      SUBRØUTINE BRHØ(T,KSPEC,NSPEC,ISIDE,XX,VLBD,SPDEN,SPDENX,RHØV,TO)
      INTEGER KSPEC,NSPEC,ISIDE
      DØUBLE PRECISION T,XX,VLBD,SPDEN(NSPEC),SPDENX(NSPEC),RHØV,TO
      IF (T .LT. TO) GØ TØ 5
      GØ TØ (101,102,103,104),ISIDE
  101 RHØV = ρ1(T,KSPEC,XX)
      RETURN
  102 RHØV = ρ2(T,KSPEC,XX)
      RETURN
  103 RHØV = ρ3(T,KSPEC,XX)
      RETURN
  104 RHØV = ρ4(T,KSPEC,XX)
      RETURN
    5 GØ TØ(1001,1002,1003,1004),ISIDE
 1001 RHØV = ρ̂1(KSPEC,XX)
      RETURN
 1002 ...
      .
      .
      RETURN
      END
```

Here ρ_i and $\hat{\rho}_i$, i=1,4 are the known boundary values.

4.4 Subroutine BRHØDT(T,KSPEC,NSPEC,ISIDE,XX,VLBD,SPDEN,SPDENX,RHØV,TO)

The variables have the same meaning as in subroutine BRHØ. This routine returns differentiated boundary values on sides with essential boundary conditions and non-differentiated values on sides with non-essential boundary conditions. Recall that if $\beta = \beta(K',ISIDE) = 0$ for species K', $1 \leq K' \leq NSPEC$, then we say that the side with index ISIDE has an essential boundary condition for species K'. If J = ISIDE, then an essential boundary condition on side J is indicated by setting the integer flag NSJ(K') = 1 in the input namelist GRID. When T < TO, this routine is being called from either a GUESS1 initial calculation or a steady-state calculation; in either case, the boundary values are constant in time. Hence if side J has an essential boundary condition and if T < TO, this routine must return RHØV = 0.DO.

If side J (J = ISIDE) has non-essential boundary conditions for species K' (NSJ(K') = 0), this routine must return undifferentiated boundary values on this side just as in subroutine BRHØ.

The common block BNDCØM must be present in this routine in order that the essential boundary indicators NS1, NS2, NS3, and NS4 are available to this

routine. This routine is used when the differentiated version (ALGBCS=F, in namelist GRID) is used. If the algebraic version is used, this routine is ignored.

The general form of this routine could be as follows.

```
      SUBRØUTINE BRHØDT(T,KSPEC,NSPEC,ISIDE,XX,VLBD,SPDEN,SPDENX,RHØV,TO)
      INTEGER KSPEC,NSPEC,ISIDE
      DØUBLE PRECISIØN T,XX,VLBD,SPDEN(NSPEC),SPDENX(NSPEC),RHØV,TO
      CØMMØN/BNDCØM/IPHASE,NS1(MAXSP),NS2(MAXSP),NS3(MAXSP),NS4(MAXSP)
      INTEGER IPHASE,NS1,NS2,NS3,NS4
      IF (T .LT. TO) GØ TØ 5
      GØ TØ (101,102,103,104),ISIDE
  101 IF (NS1(KSPEC) .NE. 1) GØ TØ 111
      RHØV = d/dt ρ1(T,KSPEC,XX)
      RETURN
  111 RHØV = ρ1(T,KSPEC,XX)
      RETURN
  102 IF (NS2(KSPEC) .NE. 1) GØ TØ 121
      .
      .
  141 RHØV = ρ4(T,KSPEC,XX)
      RETURN
    5 GØ TØ (1001,1002,1003,1004),ISIDE
 1001 IF (NS1(KSPEC) .NE. 1) GØ TØ 1011
      RHØV = 0.DO
      RETURN
 1011 RHØV = ρ̂1(KSPEC,XX)
      RETURN
      .
      .
      END
```

Here ρ_i and $\hat{\rho}_i$, i=1,4 are the same boundary value functions as in BRHØ, and MAXSP is the value used in the MORTRAN macros. Note that since the concentrations {SPDEN(K)} and the derivatives {SPDENX(K)}, evaluated on ISIDE at the position XX, are available; nonlinear boundary conditions are allowed in this subroutine.

4.5 Subroutine EXTSRC(IMATL,KSPEC,NSPEC,T,RR,ZZ,SPDEN,SPDENR,SPDENZ,VV,TO)

Variable Names and Meanings:

KSPEC....Species index.

NSPEC....Total number of species present.

T.....Value of the time.

RR.....Value of the abscissa.

ZZ.....Value of the ordinate.

IMATL....Value of the material index at the position (RR,ZZ).

SPDEN....The array {SPDEN(K') = u(K',T,RR,ZZ); K'=1,NSPEC} of concentrations.

SPDENR...The array {SPDENR(K') = $\frac{\partial u}{\partial r}$(K',T,RR,ZZ); K'=1,NSPEC} of r-direction partials of concentrations evaluated at (T,RR,ZZ).

SPDENZ...The array {SPDENZ(K') = $\frac{\partial u}{\partial z}$(K',T,RR,ZZ); K'=1,NSPEC} of z-direction partials of the concentrations evaluated at (T,RR,ZZ).

VV.....Output value of the distributed source $f_m(t,r,z,\vec{u},\overrightarrow{\nabla u})$, (m=KSPEC) appearing on the right hand side of Eq. (1.1).

TO.....Initial value of time at which a transient calculation starts.

Given KSPEC, T, RR, ZZ, IMATL, {SPDEN(K')}, {SPDENR(K')}, and {SPDENZ(K')}, this routine calculates the value of the distributed source appearing on the right hand side of Eq. (1.1). The common block BNDCOM must appear in this subroutine in order to transmit the IPHASE indicator. When IPHASE = -2, this routine is being called during a GUESS1 calculation. Recall that a GUESS1 calculation provides an initial estimate for a steady-state calculation. In this case, this routine must return an estimate for the distributed source which is independent of the concentrations {SPDEN(K')} and the gradients {SPDENR(K')} and {SPDENZ(K')}. When T < TO, a steady-state calculation is in progress. In this case the distributed source must be independent of the time T. The general form of this routine could be as follows.

```
      SUBRØUTINE EXTSRC(IMATL,KSPEC,NSPEC,T,RR,ZZ,SPDEN,SPDENR,SPDENZ,
     #     VV,TO)
      INTEGER IMATL,KSPEC,NSPEC
      DØUBLE PRECISIØN T,RR,ZZ,SPDEN(NSPEC),SPDENR(NSPEC),SPDENZ(NSPEC),
     #     VV,TO
      CØMMØN/BNDCØM/IPHASE,NS1(MAXSP),NS2(MAXSP),NS3(MAXSP),NS4(MAXSP)
      INTEGER IPHASE,NS1,NS2,NS3,NS4
      IF (IPHASE .EQ. -2) GØ TØ 10
      IF (T .LT. TO) GØ TØ 5
      VV = f(IMATL,KSPEC,T,RR,ZZ,{SPDEN(K')},{SPDENR(K')},{SPDENZ(K')})
      RETURN
    5 VV = f0(IMATL,KSPEC,RR,ZZ,{SPDEN(K')},{SPDENR(K')},{SPDENZ(K')})
      RETURN
   10 VV = f̂0(IMATL,KSPEC,RR,ZZ)
      RETURN
      END
```

Here f, f_0, and $\hat{f}_0$ are the known distributed sources, and MAXSP is the value used in the MORTRAN macros.

4.6 Subroutine FDEXTU(IMATL,KSPEC,NSPEC,T,RR,ZZ,SPDEN,SPDENR,SPDENZ, UU,UUR,UUZ,TO)

Variable Names and Meanings:

KSPEC....Species index.

NSPEC....Total number of species.

T.....Value of the time.

RR.....Value of the abscissa.

ZZ.....Value of the ordinate.

IMATL....Value of the material index.

SPDEN....Array of concentrations.

SPDENR...Array of partial derivatives with respect to r of the concentrations.

SPDENZ...Array of partials derivatives with respect to z of the concentrations.

When T, RR, and ZZ are fixed, the distributed source f, which is provided by subroutine EXTSRC, is a function of the concentrations u(1),...,u(NSPEC); the r-direction partials $u_r(1),\ldots,u_r(NSPEC)$; and the z-direction partials $u_z(1),\ldots,u_z(NSPEC)$.

UU.....The output array {UU(K') = ∂f(KSPEC)/∂u(K'); K'=1,NSPEC} of Fréchet partial derivatives of the distributed source with respect to the concentrations.

UUR.....The output array {UUR(K') = $\partial f(KSPEC)/\partial u_r(K')$; K'=1,NSPEC} of Fréchet partial derivatives of the distributed source with respect to the r-direction partial derivatives of the concentrations.

UUZ.....The output array {UUZ(K') = $\partial f(KSPEC)/\partial u_z(K')$; K'=1,NSPEC} of Fréchet partial derivatives of the distributed source with respect to the z-direction partial derivatives of the concentrations.

TO.....Initial value of time at which a transient calculation starts.

As indicated above, this routine provides the Fréchet derivatives of the distributed source. These quantities are used in the formation of the Jacobian of a nonlinear system which at each time step has to be solved by the ODE solver.

To illustrate the nature of the computations performed by this subroutine, we will consider the following example. Let NSPEC=2, and let the distributed sources be defined as follows.

$$f(1,t,r,z) = u(1,t,r,z) + u(1,t,r,z)u(2,t,r,z) + u_r(1,t,r,z)u_z(2,t,r,z)$$
$$f(2,t,r,z) = u(2,t,r,z) + (u(1,t,r,z))^2 + (u_z(2,t,r,z))^2$$

With (t,r,z) fixed, we can write these expressions as follows.

$$f(1) = u(1) + u(1)u(2) + u_r(1)u_z(2)$$
$$f(2) = u(2) + (u(1))^2 + (u_z(2))^2$$

Then for KSPEC = 1, we have:

$$UU(1) = \partial f(1)/\partial u(1) = 1 + u(2)$$
$$UU(2) = \partial f(1)/\partial u(2) = u(1)$$
$$UUR(1) = \partial f(1)/\partial u_r(1) = u_z(2)$$
$$UUR(2) = \partial f(1)/\partial u_r(2) = 0$$
$$UUZ(1) = \partial f(1)/\partial u_z(1) = 0$$
$$UUZ(2) = \partial f(1)/\partial u_z(2) = u_r(1)$$

For KSPEC = 2, we have

$$UU(1) = \partial f(2)/\partial u(1) = 2u(1)$$
$$UU(2) = \partial f(2)/\partial u(2) = 1$$
$$UUR(1) = \partial f(2)/\partial u_r(1) = 0$$
$$UUR(2) = \partial f(2)/\partial u_r(2) = 0$$
$$UUZ(1) = \partial f(2)/\partial u_z(1) = 0$$
$$UUZ(2) = \partial f(2)/\partial u_z(2) = 2u_z(2)$$

Clearly if the distributed source does not depend on the concentrations or their gradients, then this source is an external source and this routine would return arrays UU $\equiv$ 0, UUR $\equiv$ 0, and UUZ $\equiv$ 0 in this case. The general form of this routine could be as follows.

```
      SUBRØUTINE FDEXTU(IMATL,KSPEC,NSPEC,T,RR,ZZ,SPDEN,SPDENR,SPDENZ,
     #     UU,UUR,UUZ,TO)
      INTEGER IMATL,KSPEC,NSPEC
      DØUBLE PRECISIØN T,RR,ZZ,SPDEN(NSPEC),SPDENR(NSPEC),SPDENZ(NSPEC),
     #     UU(NSPEC),UUR(NSPEC),UUZ(NSPEC),TO
      CØMMØN/BNDCØM/IPHASE,NS1(MAXSP),NS2(MAXSP),NS3(MAXSP),NS4(MAXSP)
      INTEGER IPHASE,NS1,NS2,NS3,NS4
      IF (IPHASE .EQ. -2) GØ TØ 10
      IF (T .LT. TO) GØ TØ 5
      GØ TØ (101,102,...,10 NSPEC),KSPEC
  101 UU(1) = ∂f(1)/∂u(1)
      .
      .
      .
      UU(NSPEC) = ∂f(1)/∂u(NSPEC)
      UUR(1) = ∂f(1)/∂u_r(1)
      .
      .
      .
      UUR(NSPEC) = ∂f(1)/∂u_r(NSPEC)
      UUZ(1) = ∂f(1)/∂u_z(1)
      .
      .
      .
      UUZ(NSPEC) = ∂f(1)/∂u_z(NSPEC)
      RETURN
  102 UU(1) = ∂f(2)/∂u(1)
      .
      .
      .
      UUZ(NSPEC) = ∂f(2)/∂u_z(NSPEC)
      RETURN
  103
      .
      .
      .
      RETURN
    5 GØ TØ (201,202,...,20NSPEC)KSPEC
  201 UU(2) = ∂f_0(1)/∂u(1)
      .
      .
      .
20NSPEC UU(1) = ∂f_0(NSPEC)/∂u(1)
      .
      .
      .
      RETURN
   10 DØ 15 KP=1,NSPEC
      UU(KP) = 0.DO
      UUR(KP) = 0.DO
      UUZ(KP) = 0.DO
   15 CØNTINUE
      RETURN
      END
```

where MAXSP is the value used in the MORTRAN macros.

Here f and f_0 are the same distributed sources as produced by the subroutine EXTSRC. Again we emphasize that if f and f_0 are external sources, then this routine returns zeros in the arrays UU, UUR, and UUZ.

4.7 Subroutine INDATA(KSPEC,RR,ZZ,UU)

Variable Names and Meanings:

KSPEC....Species index.

RR.....Value of the abscissa.

ZZ.....Value of the ordinate.

UU.....Output value of the initial concentration for species with index KSPEC at the position (RR,ZZ).

This routine allows the user to specify the initial value of the concentrations for each species at the positions (RR,ZZ). Note that as usual, the values RR and ZZ passed to this routine are the Gaussian quadrature points used in the Galerkin approximation of the integrals. The values from this routine can be used to start either a steady-state or a transient calculation by setting INITSW = T in the input namelist GRID. The general form of this subroutine could be as follows.

```
      SUBRØUTINE INDATA(KSPEC,RR,ZZ,UU)
      INTEGER KSPEC
      DØUBLE PRECISIØN RR,ZZ,UU
      GØ TØ (101,102,...,10NSPEC),KSPEC
  101 UU = u0(1,RR,ZZ)
      RETURN
  102 UU = u0(2,RR,ZZ)
      RETURN
      .
      .
      .
      RETURN
      END
```

Here u_0(KSPEC,RR,ZZ) is the known initial concentrations. For a steady-state calculation, the user can either use this routine to provide an initial estimate, in which case he sets INITSW=T and STEDSW=T in namelist GRID; or he can use a GUESS1 calculation to form a starting estimate (GUESSW=T, INITSW=F, STEDSW=T). When INITSW=F, this routine is ignored by the program. For a transient calculation, the user again has two choices. He can use this routine if he has initial conditions that he wishes to start from. Alternatively, he performs a steady-state calculation first and uses this solution to start the transient calculation. If this latter option is to be used, then the user has the option of using this routine to start the preliminary steady-state calculation.

4.8 Subroutine RHØCP(IMATL,KSPEC,T,RR,ZZ,SPDEN,RC)

Variable Names and Meanings:

IMATL....Material index.

KSPEC....Species index.

T.....Value of time.

RR.....Value of the abscissa.

ZZ.....Value of the ordinate.

SPDEN....Array of concentrations.

RC.....Value of ρC_p returned by this subroutine.

For each species KSPEC this routine provides the coefficient $[\rho C_p]_m(t,r,z,\vec{u})$ where m=KSPEC which appears in Eq. (1.1). If $[\rho C_p]_m(t,r,z,\vec{u}) \equiv 1$ for some species index m, then the user should set the logical indicator IREVLA(m)=F, and IRHØ(m)=F, in namelist GRID. This will save on computer time since the integrals I^A in Eq. (2.35) are then precomputed. The case when $[\rho C_p]_m(t,r,z,\vec{u}) \equiv 0$ for some m is permitted in this program. In this case one sets IREVLA(m)=F, IRHØ(m)=T, and in this routine one returns RC=0.DO for the species index m. The general form of this subroutine could be as follows.

```
SUBRØUTINE RHØCP(IMATL,KSPEC,T,RR,ZZ,SPDEN,RC)
INTEGER IMATL,KSPEC
DØUBLE PRECISIØN T,RR,ZZ,RC,SPDEN(1)
RC = ρC_p(IMATL,KSPEC,T,RR,ZZ,{SPDEN(K')})
RETURN
END
```

4.9 Subroutine ANAL(KSPEC,T,RR,ZZ,VV)

Variable Names and Meanings:

KSPEC....Species index.

RR.....Value of the abscissa.

ZZ.....Value of the ordinate.

VV.....Output value.

T.....Value of the time.

This routine provides the analytic or true solution if it is known, and is used in testing the program. If the solution is unknown, the following dummy subroutine should be provided.

```
SUBRØUTINE ANAL(KSPEC,T,RR,ZZ,VV)
INTEGER KSPEC
DØUBLE PRECISIØN T,RR,ZZ,VV
RETURN
END
```

4.10 Master Driver

This is the main routine of the program and from the user's point of view serves three important functions.

(1) Any preliminary calculations can be done in a call from this routine.

(2) If the program DISPL is going to be used as a subroutine, then the call is made from this routine.

(3) In the DISPL program, the bulk of the storage is determined by the size of two arrays, AL and GPW. The size of these arrays can be set at run time in this routine.

The first function is self-evident. The second function can be elaborated on as follows. On each call to EXEC (the main subroutine), the program processes a complete problem. That is, Namelist GRID is read once and Namelist DATA is read twice. Control is then returned to the Master Driver where the user can do further calculations and repeated calls to EXEC. If the user calls EXEC when there is no further Namelist GRID input cards, the program will print

```
END ØF INPUT FILE WHILE READING NAMELIST GRID
PRØGRAM STØPPED IN EXEC
```

and the execution will be terminated. If the user calls EXEC when there are no further Namelist DATA input cards, the program will print

```
END ØF INPUT FILE WHILE READING NAMELIST DATA
```

and stop. The third function requires some discussion. First of all there are two versions of DISPL, a direct version and an iterative version. Each version has its own storage requirements with the direct version requiring more storage. Recall the following definitions:

NSPEC....Total number of species (number of partial differential equations).

KR(KZ)...Order of the B-splines in the r(z) coordinate direction.

LR(LZ)...Total number of subintervals in the r(z) coordinate direction.

INUR(I)(INUZ(J))..Vector of continuity indices in the r(z) coordinate direction. Generally INUR(I) ≡ CONTR and INUZ(J) ≡ CØNTZ with $0 \le$ CØNTR $\le$ KR-1 and $0 \le$ CØNTZ $\le$ KZ-1.

Let NR(NZ) denote the number of unknowns associated with the r(z) direction. Then

$$(4.1) \qquad NR = KR + \sum_{i=2}^{LR} (KR-INUR(i))$$

$$= KR \cdot LR - CØNTR \cdot (LR-1) \text{ when } INUR(i) \equiv CØNTR,$$

and

$$(4.2) \qquad NZ = KZ + \sum_{j=2}^{LZ} (KZ-INUZ(j))$$

$$= KZ \cdot LZ - CØNTZ \cdot (LZ-1) \text{ when } INUZ(j) \equiv CØNTZ .$$

Then the number of variables associated with each species will be

$$NRNZ = NR \cdot NZ ,$$

and the total number of variables will be

$$(4.3) \qquad NVAR = NSPEC \cdot NR \cdot NZ .$$

We now consider each of the two versions.

DIRECT VERSION

There are two matrices which dominate the storage requirements of this version both of which are stored as band matrices. For each species k, $1 \le k \le$ NSPEC, the first matrix AL is generated from the integrals I^A appearing in Eq. (2.3.4). This matrix has NRNZ rows and a band width which will be calculated. Recall that the variables are $\{U_{i,j,k}: 1 \le i \le N_R, 1 \le j \le N_Z, 1 \le k \le \text{NSPEC}\}$. Now the program stores these variables as a singly indexed array, and since there are three indices i,j,k, there are several possible choices for generating the single index used in the program. In the direct

version, k (the species index) is the most rapidly varying index. The order of the two remaining indices is then selected by the program in such a way that the bandwidth of the matrix AL is a minimum. This selection is done as follows. Let

$$\text{(4.4)} \qquad \begin{aligned} &\text{(a)} \quad \text{HORHBW} = (\text{KR}-1)+(\text{KZ}-1)\cdot\text{NR}, \quad \text{and} \\ &\text{(b)} \quad \text{VERHBW} = (\text{KR}-1)\cdot\text{NZ}+(\text{KZ}-1) \ . \end{aligned}$$

HORHBW is the half bandwidth of AL when the index i varies more rapidly than the index j, i.e. the ordering is (k,i,j) with k the most rapidly varying and j the slowest varying. VERHBW is the half bandwidth of AL when the index j varies more rapidly than i, i.e. the ordering is (k,j,i). Then if HORHBW $\leq$ VERHBW, the program selects the "horizontal" ordering (k,i,j); otherwise the program selects the "vertical" ordering (k,j,i). For each ordering, define the following parameters.

Horizontal Ordering

$$\text{(4.5)} \qquad \begin{aligned} &\text{NI} = \text{NSPEC}, \ \text{NIH} = 1, \\ &\text{NJ} = \text{NSPEC}\cdot\text{NR}, \ \text{NJH} = \text{NR}, \\ &\text{NCC} = -(1+\text{NR})\cdot\text{NSPEC}, \ \text{NCCH} = -\text{NR}. \end{aligned}$$

Vertical Ordering

$$\text{(4.6)} \qquad \begin{aligned} &\text{NI} = \text{NSPEC}\cdot\text{NZ}, \ \text{NIH} = \text{NZ}, \\ &\text{NJ} = \text{NSPEC}, \ \text{NJH} = 1, \\ &\text{NCC} = -(1+\text{NZ})\cdot\text{NSPEC}, \ \text{NCCH} = -\text{NZ}. \end{aligned}$$

In terms of these parameters, the single index n corresponding to the triple (i,j,k) is given by

$$\text{(4.7)} \qquad n = i\cdot\text{NI}+j\cdot\text{NJ}+k+\text{NCC}.$$

With this ordering for the variables $U_{ijk} = W(n)$, the matrix AL is a band matrix with NRNZ rows and a half bandwidth

$$\text{(4.9)} \qquad \text{DM} = (\text{KR}-1)\cdot\text{NIH}+(\text{KZ}-1)\cdot\text{NJH}.$$

For storage purposes, the bandwidth of AL is given by

(4.10) FBW = 3·DM+1.

This gives the storage requirement for AL as

(4.11) SNAL = FBW·NR·NZ.

If we set NAL = SNAL where SNAL is the numerical value stored in NAL, then the storage requirements for the matrix AL is set at run time with the statements appearing in Master Driver

(4.12)
```
COMMON/ALHS/AL(SNAL)
COMMON/ALSIZE/NAL
NAL = SNAL
```

The second matrix which dominates the storage requirements of the direct version is the Jacobian matrix PW used by the ODE solver GEAR. This matrix PW has NVAR rows and is stored in band form. Note that when NSPEC > 1, the matrix PW is larger than the matrix AL. The bandwidth of PW is computed as follows. With DM defined by Eq. (4.9), the half-bandwidth of PW is given by:

(4.13) MBW = (1+DM)·NSPEC-1 .

For storage purposes, the bandwidth of PW is given by:

(4.14) MFBW = 3·MBW+1 .

One might expect MFBW = 2·MBW+1; however, pivoting is required in the decomposition of PW. Therefore a factor of 3 is required rather than the factor 2. The storage requirement for PW is then determined by setting

(4.15) SNPW = MFBW·NVAR ,

and using the statements

(4.16)
```
COMMON/GEAR6/GPW(SNPW);
COMMON/PWSIZE/NPW
```

(4.17) NPW = SNPW

in Master Driver. Here SNPW is the number determined by the user from Eq. (4.15).

For the direct version of DISPL, the Master Driver should have the following basic form.

```
C THIS IS THE MASTER DRIVER FØR THE DIRECT VERSIØN ØF DISPL.
C SPACE IS ALLØCATED FUR AL AND PW HERE.  ANY PRELIMINARY
C CALCULATIØNS CAN BE DØNE IN A CALL FRØM HERE.
  DREAL AL,GPW
  CØMMØN/ALHS/AL(SNAL)
  CØMMØN/GEAR6/GPW(SNPW)
  CØMMØN/PWSIZE/NPW
  CØMMØN/ALSIZE/NAL
  NAL = SNAL
  NPW = SNPW
  CALL EXEC
  RETURN
  END
```

ITERATIVE VERSION[†]

It is clear from the discussion of the direct version that the storage requirements for the PW array can become excessive when NSPEC is large. In the direct version, the matrix PW has NVAR = NSPEC·NR·NZ rows, and the variables U_{ijk} = W(n) are ordered according to Eq. (4.7). This ordering produces a matrix PW which is banded (actually block banded with each block having dimension NSPEC). The Jacobian matrix PW is used in the ODE solver GEAR when performing the Newton-type iterations required at each time step. Hence, for each Newton iteration, a linear system has to be solved with a coefficient matrix which has the same structure as the Jacobian matrix PW. In the direct version, this coefficient matrix is factored by an LU decomposition with partial pivoting. In the iterative version, this linear system is solved by means of a Gauss-Seidel iterative procedure. However, with the ordering (4.7), the structure of PW is block banded with each block of dimension NSPEC and the bandwidth determined by the order of the splines and the mesh. With NSPEC at most 10, this structure is not convenient for block iterations since it has diagonal blocks which are too small. For this reason, a different ordering is used in the iterative version. The species index k is now the most slowly varying index while the two spatial indices are more

[†]The iterative version is not distributed at this time by the National Energy Software Center; this section can be skipped in the first reading.

rapidly varying and their relative order is again determined by selecting the order corresponding to the smaller of (HORHBW,VERHBW) which are defined in Eq. (4.4). We then define

(4.18)

$$\begin{array}{lll} & NI = 1 & \\ (a) & NJ = NR & \text{if horizontal order,} \\ & NCC = -NR & \\ & NI = NZ & \\ (b) & NJ = 1 & \text{if vertical order.} \\ & NCC = -NZ & \end{array}$$

In terms of these parameters, the single index n corresponding to the triple (i,j,k) is given by:

$$n = i \cdot NI + j \cdot NJ + NCC + (k-1) \cdot NR \cdot NZ \ . \tag{4.19}$$

With this ordering, the Jacobian matrix PW is a block matrix with each block of dimension NR·NZ. The matrix need not be block banded; however, the blocks off the diagonal are the result of coupling between species from the reaction terms and the distributed source. A block Gauss-Seidel iterative procedure is applied to the coefficient matrix generated from PW in the Newton type iterations. The number of diagonal blocks is equal to NSPEC and the dimension of each block is NR·NZ. The diagonal blocks are band matrices each having a half-bandwidth DM given by

$$DM = (KR-1) \cdot NI + (KZ-1) \cdot NJ \ , \tag{4.20}$$

where NI and NJ are given by (4.18). Since each diagonal block (one block for each species) is factored by an LU decomposition with partial pivoting, the bandwidth for storage purposes is

$$FBW = 3 \cdot DM + 1 \ , \tag{4.21}$$

and the storage per block has length FBW·NR·NZ. There are NSPEC diagonal blocks; hence the storage requirement for PW is given by

$$SNPW = FBW \cdot NR \cdot NZ \cdot NSPEC \ . \tag{4.22}$$

Thus we set

(4.23)
```
COMMON/GEAR6/GPW(SNPW)
COMMON/PWSIZE/NPW
```

(4.24)
```
NPW = SNPW
```

where SNPW for the iterative version is determined from (4.22). The storage requirement for AL in the iterative version is the same as in the direct version. Thus we have

(4.25)
```
SNAL = FBW*NR*NZ ,
```

and the common statements

(4.26)
```
COMMON/ALHS/AL(SNAL)
COMMON/ALSIZE/NAL
NAL = SNAL
```

In summary, the iterative version has the same storage requirement for AL as the direct version, but the storage requirement for PW is determined by (4.22) in the iterative version. The basic form of the Master Driver is the same as the form given for the direct version.

5. DESCRIPTION OF USER-SUPPLIED DATA

For the computational phase of DISPL, the program requires the user-supplied subroutines discussed in section 4, certain Mortran macro input, unit assignments, and input data. In this section we discuss this additional information as well as the corresponding needs of all three of the optional graphics programs. In section 5.1 we discuss the Mortran macros and unit assignments for the computational and graphics programs in DISPL. The namelist input for the computational phase is discussed in section 5.2 and 5.3, i.e. namelists GRID and DATA. The graphics namelists are discussed in sections 5.4-5.7. Specifically, namelist FORMAT is used in all three graphics packages and is discussed in section 5.4. Namelist CSPIN (section 5.5) provides the remaining input for the cross-section plot package; namelist CNTRIN (section 5.6) provides the remaining data for the contour graphics package; and namelist DIM3IN (section 5.7) provides the remaining data for the three-dimensional perspective plotting package. The only user-supplied subroutine in the graphics packages occurs in the cross-section plotting package. In this program the user can plot the analytic solution and, in this case, he must provide a single precision version of subroutine ANAL.

In section 5.8 we discuss restart procedures used in the computational phase of the code. Finally, in section 5.9 we illustrate Job Control Language (JCL) instructions that can be used in executing the DISPL package on an IBM 370/195, and also describe the organization of the tape containing the DISPL code.

Before discussing the macros, unit assignments, and namelists, we begin with a general discussion of the namelist feature for users unfamiliar with it. For concreteness we discuss namelist feature in connection with the input data for the computational phase of the code.

For each problem, three sets of input data cards are required. Recall that a complete problem may consist of:

(a) a steady-state calculation, or

(b) a transient calculation, or

(c) a steady-state followed by a transient calculation.

Consider the last situation. We divide the input data into three sections.

The first section consists of that data which is unchanged in passing from a steady-state to a transient calculation (e.g. specification of the domain including the approximation grid and interfaces, number of species, order of the approximating piecewise polynomials, etc.). The second section consists of the remaining data necessary to specify the problem (e.g. boundary values, gap data, reaction coefficients, essential side indicators, output control, etc.). Some or all of the data in the second section can be changed just prior to a transient calculation by resetting this data in the third input data section. Thus, for example, in situation (c), the program uses the first two sections of input data for the steady-state calculation, and the first two sections subject to modifcations in the third section for the transient calculation.

These three sections of input data are communicated to the program via the Namelist feature of Fortran. Namelist uses a free-format similar to ordinary assignment statements and is described in [13, pp. 54-55]. In Namelist terms, the first input section is called GRID, while the other two are called DATA. The data cards thus have the form

```
&GRIDb free-format statements
free-format statements
     .
     .
     .
   &END
&DATAb free-format statements
free-format statements
     .
     .
     .
   &END
&DATAb free-format statements
free-format statements
     .
     .
     .
   &END
```

The "&GRID" and "&DATA" must begin in column 2 and must be followed by a blank, i.e. the "b" indicated above. All other cards may begin in any column other than column 1. The "&END" can be on a separate card or at the end of the last free-format card. The free-format cards have the following form.

```
A=1.,B=2.,3.,B(3)=4.,
```

where, in this case, A is a variable set to 1. and the first three elements of the vector B are set to 2., 3., and 4., respectively. Any number of blank columns can follow each comma. Thus, one convenient way of using the namelist format is to put only one variable or vector component on a card, e.g.

```
A=1.,
B=2.,
  3.,
B(4)=4.,
```

and so allow for easy modification of the input cards. It is recommended that two- and three-dimensional data be entered with explicit subscripts for readability.

The three namelist groups must be provided even if they are not used, i.e. even if there are no variables to be read in, the "&DATA" and "&END" must be provided for the third set of data cards. The variables within a given namelist input can be in any order.

5.1 Mortran Macros and Unit Assignments

Before describing these namelist variables we discuss some other inputs the user must provide which are referred to in the definition of these variables. First, the computational phase of the program has the ability to dump information on unit 11 which can be used for later restart calculation. The restart reads data from unit 10 and when it dumps, the dump is on unit 11. Notice that this procedure will allow multiple restarts from a fixed set of dump information. The user must therefore assign units 10 and 11 (to disk or tape datasets). The program also writes data on unit 12, optionally, for use in later graphical analysis. Again, the user must assign this unit. The assignment of units 10, 11 and 12 must be done even if the program does not use these units.

One of the advantages of using MORTRAN is that the size of storage arrays are specified in macro variables. These variables and many of the common blocks in the code are given in a macro dataset located on the first file of the tape of the code. The macro dataset is assigned to unit 2 during compilation. (The computational code itself is on the second file, the B-spline package is on the third file, and the graphics programs are on the fourth through sixth files.) By changing the values of the following macro integer variables, the Fortran code produced by the Mortran processor will be correspondingly changed.

MAXBRK....integer macro for the maximum number of breakpoints in either the r or z direction. In either direction this number is a bound on the sum of the interior mesh points, interface points,

and end points.

MAXTQD....integer macro for the maximum total number of quadrature points per mesh interval in either the r or z direction.

MAXSP.....integer macro for the maximum number of species.

MAXGAP....integer macro for the maximum number of gap interfaces in either coordinate direction.

MAXK......integer macro for the maximum order of splines in either direction.

MAXNOT....integer macro for the maximum number of spline knots in either direction.

MXNRNZ....integer macro for the maximum number of variables permitted for any single species.

MXNVAR....integer macro for the maximum number of total variables.

MXRGRD....integer macro for the maximum number of points in the r direction for the user-specified grid.

MXZGRD....integer macro for the maximum number of points in the z direction for the user-specified grid.

INFILE....integer macro for the standard input unit.

DUMPRD....integer macro for the unit from which a restart is read.

DUMPWR....integer macro for the unit on which a dump is written.

GRAPHWR...integer macro for the unit on which the unformatted graphics information is written.

At the time of compilation, these macros should be selected with some care since they determine some of the core requirements of the code. To fix our attention, consider the following choices for these macros.

MAXBRK=30, MAXTQD=4, MAXSP=2, MAXGAP=2, MAXK=4, MAXNOT=40, MXNRNZ=100, MXNVAR=100, MXRGRD=20, MXZGRD=20, INFILE=5, DUMPRD=10, DUMPWR=11, GRAPHWR=12.

With these values, and the choice NPW=NAL=784 in the common statements for AL and GPW appearing in the Master Driver routine (see section 4.10), the direct version requires 362K bytes of machine memory on an IBM 370/195 computer. In order to judge the effect of changing these macros, we list the arrays which depend on these macros.

Real arrays

A(MAXK,MAXTQD,MAXBRK), APRIM(MAXK,MAXTQD,MAXBRK), B(MAXK,MAXTQD,MAXBRK), BPRIM(MAXK,MAXTQD,MAXBRK)

AINT(MAXK,MAXBRK), BINT(MAXK,MAXBRK)

AAINT(MAXK,MAXK,MAXBRK), BBINT(MAXK,MAXK,MAXBRK)

HVGAP(MAXGAP,MAXBRK,MAXSP), HHGAP(MAXGAP,MAXBRK,MAXSP)

AAINT(MAXK,MAXK,MAXBRK), BBINT(MAXK,MAXK,MAXBRK)

HVGAP(MAXGAP,MAXBRK,MAXSP), HHGAP(MAXGAP,MAXBRK,MAXSP)

HU1(MAXBRK,MAXSP), HU2(MAXBRK,MAXSP), HU3(MAXBRK,MAXSP), HU4(MAXBRK,MAXSP)

H1(MAXBRK,MAXSP), H2(MAXBRK,MAXSP), H3(MAXBRK,MAXSP), H4(MAXBRK,MAXSP)

RGAU(MAXTQD,MAXBRK), ZGAU(MAXTQD,MAXBRK), R2PI(MAXTQD,MAXBRK)

CK(MAXSP,MAXSP), CKK(MAXSP,MAXSP,MAXSP)

ALPHA(MAXSP,4), BETA(MAXSP,11), GAMMA(MAXSP,11) BRK)

R(MAXBRK), Z(MAXBRK), WR(MAXTQD), WZ(MAXTQD)

HSIG(MAXBRK), HTAU(MAXBRK), RG2PI(MAXBRK), RIF(MAXBRK), ZIF(MAXBRK), RMESH(MAXBRK), ZMESH(MAXBRK)

RKNOT(MAXNOT), ZKNOT(MAXNOT)

RGRID(MAXRGRD), ZGRID(MXGRD)

Integer arrays

IL(MAXBRK), JL(MAXBRK)

SVGAP(MAXBRK), THGAP(MAXBRK)

MLTAB(MAXBRK,MAXBRK), MATL(MAXBRK,MAXBRK)

IFTYPR(MAXBRK), IFTYPZ(MAXBRK)

NS1(MAXSP), NS2(MAXSP), NS3(MAXSP), NS4(MAXSP)

IGDSIG(MXRGRD), JGDTAU(MXZGRD)

We now consider the unit assignments and macro variables associated with the graphics programs. In all three cases the graphics programs read the graphics dataset created on unit 12 by the computational phase of DISPL. For a particular graphics run this dataset must be again assigned to unit 12. The graphics programs are all written in Mortran and utilize the same macro dataset as was used in the computational phase. This dataset must be assigned to unit 2. Except for the assignment of graphics JCL [9] there are no other units required for the cross-section plot (CSP) or the three-dimensional perspective (THREED) programs. However, the contour program (CØNTØR) does require additional information. This program generates a dataset on unit 13 (and must therefore be assigned) which is in a format acceptable for the

CONTOUR.BLACKBOX program available at ANL [11]. The BLACKBOX program is then executed as a second job step after CØNTØR has completed its execution.

The Macro variables used in a given execution of a graphics program must be compatible with the Macros used in the computational phase of DISPL. That is, certain Macro variables used in generating the graphics dataset on unit 12 during the computational phase must have the same value when used in a graphics execution. These variables are MAXSP, MAXK, MXNRNZ, MXNVAR, MAXBRK, and MAXNOT. In addition, other Macro variables must be set. The Macro dataset which is on the first file of the tape is designed to simplify this task. This dataset is compatible with all of the graphics programs as well as the computational phase of DISPL. Thus, by using the same version of this dataset for all of the programs in DISPL we assure the compatibility of MAXSP, MAXK, MXNRNZ, MXNVAR, MAXBRK, and MAXNOT. The following Mortran variables are used only by the graphics programs and must be set for a given compilation of a graphics program.

MAXFRM....integer Macro for the maximum number of duplicate graphs to be created by the CSP program. These duplicate frames are generated only in CSP and only in the cinema mode.

MAXINT....integer Macro which must be set to twice MAXBRK. This number is only used in the CSP program.

NRES1.....integer Macro for the maximum number of points to be plotted on curves in the CSP program. Notice that this value is an upper bound on the Namelist FORMAT variable NRESIN when used in the CSP program.

MRES......integer Macro which must be set to NRES1+2. This value is used only in CSP.

NRESD.....integer Macro for the maximum number of grid points used along each axis in the CØNTØR and THREED programs. This value is used only in those two programs and is an upper bound on NRESIN when used in connection with these programs.

NWORK.....integer Macro for the size of a workspace array used only in the THREED program. NWORK should be 2*NRESD+4.

MXGNUM....integer Macro for the maximum number of time plots generated by CØNTØR or THREED. This macro is used only in those two programs and is associated with arrays which are used only if ITIME in Namelist CNTRIN or DIM3IN is true. In this situation MXGNUM is an upper bound on the Namelist FORMAT variable IGNUM.

MXGRP.....integer Macro for the maximum number of frames to be plotted for a given time value in the CSP program.

5.2 Namelist GRID

We next describe the input variables in GRID and their default values, i.e. values used if not specified.

KR.....integer variable for the order of the spline approximation in the r-direction.
KR=4, (Default value) ($1 \leq KR \leq MAXK \leq 12$)

KZ.....integer variable for the order of the spline approximation in the z-direction.
KZ=4, ($1 \leq KZ \leq MAXK \leq 12$)

NQR.....integer variable for the order of quadrature formula used for r-direction integration.
NQR=MAX(KR,1) ($1 \leq NQR \leq MAXTQD \leq 12$)

NQZ.....integer variable for the order of quadrature formula used for z-direction integration.
NQZ=MAX(KZ,1) ($1 \leq NQZ \leq MAXTQD \leq 12$)
However, these are conservative choices. The choices NQR=KR-1 and NQZ=KZ-1 seem to be optimal. See Sample problem 7.4 for a further discussion.

NSPEC.....integer variable for the total number of chemical species.
NSPEC=1, ($1 \leq NSPEC \leq MAXSP$)

DELTA.....integer variable for geometry selection. Use 0 for rectangular geometry, 1 for cylindrical geometry, and 2 for spherical geometry (one dimension only).
DELTA=0,

RLOW.....real variable for the left hand boundary of the domain.
RLOW=0.0,

RUP.....real variable for the right hand boundary of the domain.
RUP=1.0,

ZLOW.....real variable for the lower boundary of the domain.
ZLOW=0.0,

ZUP.....real variable for the upper boundary of the domain.
ZUP=1.0,

NTIR.....integer variable for the total number of interfaces along the R axis.
NTIR=0, ($0 \leq$ NTIR $\leq$ MAXBRK)

NTIZ.....integer variable for the total number of interfaces along the Z axis.
NTIZ=0, ($0 \leq$ NTIZ $\leq$ MAXBRK)

RIF.....real vector of interface mesh points (R coordinate).
RIF=0.0,0.0,... (vector is initialized to zero) (Dimension is NTIR)

ZIF.....real vector of interface mesh points (Z coordinate).
ZIF=0.0,0.0,... (vector is initialized to zero) (Dimension is NTIZ)

IFTYPR....interface type indicator integer for r-direction. Use 0 for a gap interface, or 1 for a continuous interface. (Notice that all r-direction interfaces at a particular r-coordinate are therefore of the same type.)
IFTYPR=1,1,... (Dimension is NTIR)

IFTYPZ....interface type indicator integer for z-direction. Use 0 for a gap interface or 1 for a continuous interface. (Notice that all z-direction interfaces at a particular z-coordinate are therefore of the same type.)
IFTYPZ=1,1,... (Dimension is NTIZ)

NMR.....integer variables of the total number of non-interface mesh points in the r-direction (not including the end points RLOW and RUP).
NMR=0, (0 < NMR < MAXBRK)

NMZ.....integer variable of the total number of non-interface mesh points in the z-direction (not including the end points ZLOW and ZUP).
NMZ=0, ($0 \leq$ NMZ $\leq$ MAXBRK)

RMESH.....real vector of non-interface mesh points (r-coordinate). If the vector RMESH is omitted, the code will use NMR to generate a set of equally spaced interior points.
RMESH(I), I=1,NMR

ZMESH.....real vector of non-interface mesh points (z-coordinate). If the vector ZMESH is omitted, the code will use NMZ to generate a set of equally spaced interior points.
ZMESH(J), J=1,NMZ

CØNTR.....integer variable of continuity (in the r-direction) across mesh points. ($0 \leq$ CØNTR $\leq$ KR-1)
CØNTR=KR-1, (the default value is set to KR-1)

CØNTZ.....integer variable of continuity (in the z-direction) across mesh points. ($0 \leq$ CØNTZ $\leq$ KR-1)
CØNTZ=KR-1,

INUR.....integer vector of continuity indices (in the r-direction) at non-interface mesh points. These values can be used to override the continuity value CØNTR at specified non-interface mesh points.
INUR=CØNTR,CØNTR,... (the default value of CØNTR is set by the code) (Dimension is NMR)

INUZ.....integer vector of continuity indices (in the z-direction) at non-interface mesh points. These values can be used to override the continuity value CØNTZ at specified non-interface mesh points.
INUZ=CØNTZ,CØNTZ,... (Dimension is NMZ)

MATL.....two-dimensional integer array of material indices. MATL is dimensioned (MAXBRK,MAXBRK) and is defined for

MATL(I,J) I=1,...,NTIR+1
J=1,...,NTIZ+1

The default is MATL=1,1,...

ALGBCS....logical indicator for selecting method of treating algebraic boundary conditions. If ALGBCS is true, the values of the boundary conditions as provided through the user-supplied subroutine BRHØ are used. If ALGBCS=F, the user routine BRHØDT is used to provide the boundary conditions. Note that in BRHØDT one must provide the time derivative of the essential boundary values. If ALGBCS=F, BRHØ can be a dummy routine, while if ALGBCS=T, BRHØDT can be a dummy routine.
ALGBCS=T,

CØNSRV....logical indicator of selecting conservative or substantial derivative form of convection term. If CØNSRV is true, the conservative form is used. If CØNSRV is false the substantial derivative form is used.
CØNSRV=T,

IREVLA....vector of logical variables for indicating the presence of a heat capacity coefficient in the time derivative term. IREVLA has dimension NSPEC and IREVLA(K)=T, indicates that $[\rho C_p]_K(T,R,Z,\vec{U}) \not\equiv 1.0$; in this case the left hand side of Eq. (3.1) is reevaluated at each time step unless IRHØ(K)=T.
IREVLA=F,F,...

IRHØ.....vector of logical variables for indicating whether $[\rho C_p]_K \equiv 0$ or not. IRHØ(K)=T, implies that $[\rho C_p]_K \equiv 0$. IRHØ(K)=F, implies that $[\rho C_p]_K \not\equiv 0$. See Note at end of namelist GRID (page 78).
IRHØ=F,F,...

INITSW....logical indicator for using the user-supplied subroutine INDATA to provide an initial estimate of the solution for the steady-state calculation or initial conditions for the transient calculation.
INITSW=T,

GUESSW....logical indicator for using the program's initial estimate of the solution for the steady-state calculation. Whenever possible the user should provide an initial estimate for a steady-state calculation.
GUESSW=T,

STEDSW....logical indicator for doing a steady-state calculation.
STEDSW=T,

TRANSW....logical indicator for doing a transient calculation. A transient calculation expects initial conditions. These can be provided by either the initial spline coefficients, a least squares fit to the initial data (INITSW=T,) or a steady-state calculation.
TRANSW=T,

ISTDFQ....integer variable for the frequency of output from the steady-state calculation. Output occurs every ISTDFQ time steps after the first time step. Output also occurs on the first time step. (Notice that ISTDFQ $\leq$ 0 is meaningless.) The output includes an evaluation of the approximate solution at the Gaussian point grid (or optionally on user-specified grid) and may also give the error at points on the grid via calls to user-supplied subroutine ANAL.
ISTDFQ=10,

IRGRD.....integer variable of the number of points in the r-direction for the optional user-specified output grid. If IRGRD=0 then the tensor product of the Gaussian points in the r and z direction is used to form the output grid. If IRGRD is positive, then the grid used is the tensor product of RGRID and ZGRID (see below). For a one-dimensional problem in z, set IRGRD=1,.
IRGRD=1, (0 $\leq$ IRGRD $\leq$ MXRGRD)

RGRID.....real vector of IRGRD points in the r-direction for user's specified grid. When IRGRD=0 or 1, the vector RGRID is not needed. When IRGRD $\geq$ 2, the vector RGRID can be omitted. In this case, the code generates a set of IRGRD equally spaced output points RGRID(I) such that the first point is equal to RLØW and the last point is equal to RUP.

JZGRD.....integer variable of the number of points in the z-direction for user's specified grid. For a one-dimensional problem in r, set JZGRD=1,.
JZGRD=1, (0 $\leq$ JZGRD $\leq$ MXZGRD)

ZGRID.....real vector of JZGRD points in the z-direction for user's specified grid. When JZGRD=0 or 1, ZGRID is not needed. When JZGRD $\geq$ 2, ZGRID can be omitted in which case the code generates equally spaced output points, the first and last being ZLØW and ZUP.

IORD.....integer vector indicating order of derivatives computed on the user-supplied output grid. The code always provides the approximate solution (either user-supplied or Gaussian). However, if a user-supplied grid is used, partial derivatives of the approximation can be requested. IORD is a vector of, at most, fifteen components. Each component is an integer of the form 10*I+J and will cause the mixed partial of order I with respect to R and order J with respect to Z to be computed. Thus IORD = 10,11, will cause $\frac{\partial U(K)}{\partial R}$ and $\frac{\partial U(K)}{\partial R \partial Z}$ to be computed as well as U(K) for all K. Notice that the user need not specify I=J=0 (the function value case).
IORD=0,0,...

IANAL.....logical indicator for giving the analytic solution in user-supplied subroutine ANAL. If IANAL=F, subroutine ANAL can be a dummy routine. If IANAL=T, subroutine ANAL is used to compute the steady state or the transient solution on the Gaussian or user-specified grid.
IANAL=F,

ISTDRS....logical indicator for restarting the steady-state calculation from a dump. This indicator and the next two indicators can be used to restart a calculation from a dump generated during a previous run. The code dumps information under the following four conditions: if it detects that time is expiring during a steady-state calculation; at a normal conclusion of a steady-state calculation; if it detects that time is expiring during a transient calculation; and at the normal conclusion of a transient calculation. Only the last dump during a given run is meaningful since each dump writes from the beginning of the file. Thus to generate a dump at the normal conclusion of the steady-state calculation, set TRANSW=F. (See §5.8 for further

information about restarting.) The ISTDRS indicator is associated with restarting the first type of dump, the DUMPSW indicator is associated with restarting the second type; and the ITRARS indicator is associated with the third type. The fourth type of dump can be utilized as follows: Set up an array of UTØUT of dimension N2 and set NUTØUT = N1<N2. The program will then dump at T=UTØUT(N1). On the restart run set INITSW, GUESSW and STEDSW to false and TRANSW to true and use NTØUT=N2. The calculation will restart from UTØUT(N1) and process to UTØUT(N2). To restart a steady-state calculation set ISTDRS=T, INITSW=F, GUESSW=F, and STEDSW=T. The code will then read the current value of the approximating coefficients and time (variables W and T, respectively) from unit 10.
ISTDRS=F,

DUMPSW....logical indicator for reading coefficients from unit 10. If the dump in a previous run occurred at the normal conclusion of the steady-state calculation, the code dumped the final coefficients. By setting DUMPSW=T, these coefficients can be read from unit 10 and can be used in either a new steady-state or transient calculation.
DUMPSW=F,

ITRARS....logical indicator for restarting a transient calculation which ran out of time. If ITRARS=T, then the current value of the approximate coefficients and time (variables W and T, respectively) are read from unit 10. When ITRARS is true, INITSW, GUESSW, and STEDSW must be false and TRANSW must be true.
ITRARS=F,

Note: If $[\rho C_p](K) \equiv 1$ use IREVLA(K)=F, and IRHØ(K)=F, (both default values).
If $[\rho C_p](K) \equiv 0$ use IREVLA(K)=T, and IRHØ(K)=T, (IREVLA(K)=F, and IRHØ(K)=T, can also be used -- it generates a warning message).
If $[\rho C_p](K) \not\equiv 1$ and $\not\equiv 0$ use IREVLA(K)=T, and IRHØ(K)=F,.

5.3 Namelist DATA

As mentioned earlier there are two sets of Namelist DATA. The first set is read once immediately after GRID before any calls to the calculation drivers. The second set is read just before the call to TIMEX, the transient driver. Thus the first set is read before either a steady state or a transient calculation is started, while the second set is read just before a transient calculation is started. If a steady-state calculation is to be followed by a transient calculation, the first set is used to define the steady-state case while the second set is used to define the transient case. If there are no changes in the variables appearing in the first set of Namelist DATA, then the second set can be a null set. Furthermore, the second set need only contain those variables in Namelist DATA which are changed when passing from a steady-state to a transient calculation. The variables and defaults for Namelist DATA are as follows.

W.....real vector of spline coefficients. W is an array of dimension MXNVAR. This array can be used to specify the initial values for either a steady-state or a transient calculation by setting INITSW=F, in Namelist GRID. However, since W is an array of spline coefficients, it can be specified only in exceptional circumstances (cf. sample problem 7.4). Moreover, in order to specify W one must know the ordering of W used in the code. This ordering is discussed in section 4.10.
W=1.0,1.0,...,

ALPHA.....real variable array of boundary value coefficients. ALPHA is dimensioned (MAXSP,4) and the user must provide

```
ALPHA(I,J)  I=1,2,...,NSPEC
            J=1,2,3,4 (side index)
```

The default is ALPHA=0.0,0.0,...,

BETA.....real variable array of boundary value coefficients. BETA is dimensioned (MAXSP,4) and the user must provide

```
BETA(I,J)  I=1,2,...,NSPEC
           J=1,2,3,4 (side index)
```

The default is BETA=1.0,1.0,...,

GAMMA.....real variable array of boundary value coefficients. GAMMA is dimensioned (MAXSP,4) and the user must provide

GAMMA(I,J) I=1,2,...,NSPEC
J=1,2,3,4 (side index)

The default is GAMMA=0.0,0.0,...,

NS1(K)....integer indicator for essential boundary conditions of side 1 for K-th species. If NS1(K)=1 then the boundary condition is essential, i.e. BETA(K,1)=0. If NS1(K) is 0 then the boundary condition on side 1 is non-essential for species K. When NS1(K)=-1, no boundary condition is imposed on side 1 for species K.†
NS1=0,0,...

NS2(K)....integer indicator for essential boundary conditions on side 2 for the K-th species. See description of NS1 indicator.
NS2=0,0,...

NS3(K)....integer indicator for essential boundary conditions on side 3 for the K-th species. See description of NS1 indicator.
NS3=0,0,...

NS4(K)....integer indicator for essential boundary conditions on side 4 for the K-th species. See description of NS1 indicator.
NS4=0,0,...

For the standard Dirichlet or Newmann condition see the variables DRCHLT and NEUMAN at the end of this namelist.

HU1.....real array of boundary function h for side 1. HU1 is dimensioned (MAXBRK,MAXSP) and the user must provide

HU1(I,K) I=1,2,...,NTIZ+1
K=1,2,...,NSPEC

The default value is HU1=1.0,1.0,...,

HU2.....real array of boundary function h for side 2. HU2 is dimensioned (MAXBRK,MAXSP) and the user must provide

HU2(I,K) I=1,2,...,NTIR+1
K=1,2,...,NSPEC

The default value is HU2=1.0,1.0,...,

†When specifying the indicators NS1(K), NS2(K), NS3(K), and NS4(K) in Namelist DATA (pp. 79-80), the user must be sure to also specify the corresponding values of ALPHA(K,I), BETA(K,I), and GAMMA(K,I). These values can be specified directly or by use of the logical indicators DRCHLT(K,I) or NEUMAN(K,I).

HU3.....real array of boundary function h for side 3. Description is as for HU1.

HU3=1.0,1.0,...,

HU4.....real array of boundary function h for side 4. Description is as for HU1.

HU4=1.0,1.0,...,

HVGAP.....real array of gap coefficients in the z-direction. HVGAP is of dimension (MAXSP,MAXBRK,MAXBRK) and the user must provide

```
HVGAP(K,I,J)  K=1,2,...,NSPEC
              I=1,2,...,number of vertical gaps along R
              J=1,2,...,NTIZ+1
```

Default value is HVGAP=0.0,0.0,...,

HHGAP.....real array of gap coefficients in the r-direction. HHGAP is of dimension (MAXSP,MAXBRK,MAXBRK) and the user must provide

```
HHGAP(K,J,I)  K=1,2,...,NSPEC
              J=1,2,...,number of horizontal gaps along Z
              I=1,2,...,NTIR+1
```

Default value is HHGAP=0.0,0.0,...,

CK.....real array of first order reaction coefficients. CK is of dimension (MAXSP,MAXSP) and the user must supply

```
CK(K',K)  K'=1,2,...,NSPEC
          K=1,2,...,NSPEC
```

where CK(K',K) is the first order reaction coefficient from the K'-th species into the K-th species. The terms CK(K',K)*u(K') are summed over K' to form the total first order reaction source into the K-th species. The default value is

CK=0.0,0.0,...,

CKK.....real array of second order reaction coefficients. CKK is of dimension (MAXSP,MAXSP,MAXSP) and the user must supply

```
CKK(I,J,K)  I=1,2,...,NSPEC
            J=1,2,...,NSPEC
            K=1,2,...,NSPEC
```

where CKK(I,J,K) is the second order reaction coefficient from the I-th and J-th species into the K-th species. The terms

CKK(I,J,K)*U(I)*U(J) are summed over I and J to form the total second order reaction source for the K-th species. The default value is

CKK=0.0,0.0,...,

TO.....real variable for initial time at start of transient calculation.
TO=0.0,

NUTOUT....integer variable related to the frequency of output during the transient computation. NUTOUT is the number of major time values in the array UTOUT (see below).
NUTOUT=2,

UTOUT.....real vector of NUTOUT major time values. Note that UTOUT(1) is the time at which the transient calculation begins and must agree with TO, the physical initial time. UTOUT is of dimension 100 and its default value is

UTOUT=0.0,1.0,2.0,0.0,0.0,...,

NUFREQ....integer variable of the number of equally spaced points on each interval [UTØUT(I),UTØUT(I+1)]. Output will occur at each such point. If NUFREQ=1 then output will occur at each time UTØUT(I), $1 \leq I \leq$ NUTØUT. Notice that NUFREQ ≤ 0 is meaningless.
NUFREQ=1,

GRAPH.....logical indicator to write output of unit 12 for later use in graphic analysis. This output will occur during each ordinary output for the steady-state or transient calculation.
GRAPH=F,

NUMGRF....number of run. This integer variable is written on unit 12 and is used by the graphics programs to identify the run which produced the graphics file.
NUMGRF=0,

EPS.....real variable of accuracy requirement for the GEAR subroutines used in solving the ordinary differential equations system.
EPS=.0001,

HINIT.....real variable of initial step size for GEAR calculation.
HINIT=.001,

MXGORD....integer variable for selecting the maximum order used in the ODE solver. MXGORD must be between 1 and 5.
MXGORD=5,

IPRSW1....integer variable for certain additional output from subroutine INPROC. If IPRSW1=0 print will not occur. If IPRSW1=1 the print will occur. (This print switch and the following print switches are primarily of use for debugging the code.)
IPRSW1=0,

IPRSW2....integer variable for additional output from subroutine INIFIT and certain output from subroutine INPROC. (See description of IPRSW1.)
IPRSW2=0,

IPRSW3....integer variable for additional output from subroutine GFUN and certain output from subroutine PEDERV. (See description of IPRSW1.)
IPRSW3=0,

IPRSW4....integer variable for additional output from subroutines GUESS1, RHS, and TIMEX. (See description of IPRSW1.)
IPRSW4=0,

IPRSW5....integer variable for certain additional output from subroutine PEDERV. (See description of IPRSW1.)
IPRSW5=0,

IPRSW6....integer variable for additional output from subroutine BLKSOR (in the iterative version of the code only).
IPRSW6=0,

DRCHLT(K,I)..logical indicator specifying that the K-th species on side I satisfies a Dirichlet boundary condition. DRCHLT(K,I)=T, will set NSI(K)=1, ALPHA(K,I)=1.D0, BETA(K,I)=0.D0, and GAMMA(K,I)=1.D0.
DRCHLT(K,I)=F,

NEUMAN(K,I)..logical indicator for a Neumann boundary condition. NEUMAN(K,I)=T, will set NSI(K)=0, ALPHA=0.D0, BETA=1.D0, and GAMMA=1.D0.
NEUMAN(K,I)=F,

5.4 Namelist FØRMAT

This and the following three namelists provide the input for the graphics programs which are available with this code. The first namelist, FØRMAT, provides variables common to all three of the graphics packages. The first graphics package is CSP which provides a one-dimensional cross section plotting capability. The namelist for this package is CSPIN. The second graphics package is CØNTØR which provides a contour plotting capability. The namelist for this package is CNTRIN. The third graphics package is THREED which provides perspective surface plotting capability. The namelist for this package is DIM3IN.

The graphics programs will process files that were generated by several calls to EXEC. For each complete problem being graphed (that is for each time EXEC was called), the graphics program reads Namelist FØRMAT and either Namelist CSPIN, CNTRIN, or DIM3IN. For example, if EXEC was called twice and we are using the CSP program, we need Namelist FORMAT and CSPIN followed by Namelist FØRMAT and CSPIN.

In discussing the namelist input for these graphics packages, the following terminology will be used. A frame is a physical plotting area, and for each time value one can have several frames. In the CØNTØR and THREED package there is just one coordinate axis per frame and one curve (contour, surface plot) per axis. In the CSP package, one can have one or more coordinate axis per frame with the same number of axis in each frame. If there is more than one axis per frame, then there can be only one curve per axis. On the other hand if there is one coordinate axis per frame, then there can be several curves per axis.

The following variables are in namelist FØRMAT.

ITRTV.....integer variable indicator specifying which version of the code was used to generate to graphics dataset (unit 12). This indicator establishes the ordering assumed for the spline coefficients. If ITRTV is 1, the ordering assumed is that used in the iterative version. If ITRTV is anything else, the ordering is assumed to be that used in the direct version of the code.
ITRTV=0,

IGNUM.....integer variable for the number of curves (or contours or three-dimensional perspectives) to be produced at each time value.
IGNUM=1, ($1 \leq$ IGNUM $\leq$ MXGNUM)

NRESIN....integer variable for the number of grid points used for graphical purposes in each coordinate direction. For CSP it

is recommended to use 200 $\leq$ NRESIN $\leq$ 1000. For CØNTØR and THREED recommended values of 10 $\leq$ NRESIN $\leq$ 50.
NRESIN=NRES1, for CSP (1 $\leq$ NRESIN $\leq$ NRES1)
NRESIN=NRESD, for CØNTØR or THREED (1 $\leq$ NRESIN $\leq$ NRESD)

ITIME.....logical variable for use in time plots. When ITIME=T, the graphs generated will involve time as one of the independent variables. In the case of CSP, the solution at a given point is plotted as a function of time. At present this plot consists only of points without any connecting curve. In the case of CØNTØR, contours are plotted on a plane in which the horizontal axis is time, and the other axis is a line parallel to either the r or the z axis. This line is specified by setting RMIN,ZMIN,RMAX,ZMAX in Namelist CNTRIN. In the case of THREED, the independent variables are the same as in CØNTØR, however, the coordinates of the line are specified in Namelist DIM3IN rather than CNTRIN. Note: for CØNTØR and THREED, the time values must be equally spaced.
ITIME=F, (Default value)

NTIME.....integer variable for the number of time values used when ITIME=T. When ITIME=F, this variable can be ignored.
NTIME=NRES1, (Default for CSP) (1 $\leq$ NTIME $\leq$ NRES1)
NTIME=NRESD, (Default for CØNTR and THREED) (1 $\leq$ NTIME $\leq$ NRESD)

5.5 Namelist CSPIN

The variables described here are used in the cross section plotting (CSP) program only.

ICN.....integer variable indicator for cinema mode. If ICN=1 the code uses its cinema mode to produce multiple copies of each frame during graphing. If ICN is anything else, cinema mode is not used and only one copy of each frame may be requested.
ICN=0,

IFØRMT....integer indicator for grouping format of graphs. If IFØRMT is set to 1, the graphs will be produced in packed format. That is, all the curves in a frame will be plotted on a single set of axes. If IFØRMT is set to anything else, the code will use its separate mode. In this mode each curve on a given frame will be plotted on a separate set of axes. Thus multiple sets of axes will occur on a single frame.
IFØRMT=0,

IANAL.....logical indicator for graphing the analytic solution. If IANAL is TRUE, the analytic solution for each species is plotted along with the numerical solution. In this case the user must supply a SINGLE PRECISION version of SUBROUTINE ANAL. If IANAL is FALSE the analytic solution will not be plotted although a dummy version of SUBROUTINE ANAL must be provided.
IANAL=F,

YAXMIN....real vector used as an estimate of the minimum value of the ordinate for all curves in a given frame (where the frame number for a given time is defined in LGRØUP). This estimate is checked by the CSP code and is used if it is reasonably close to the actual minimum value needed for all the curves within the given frame. By proper selection of YAXMIN and YAXMAX a constant vertical axis range can be used for all time values in a frame, e.g. for movie generation. Since there is a separate estimate for each frame there are $\max_{I}$(LGRØUP(I)) components in YAXMIN and YAXMAX.
YAXMIN=0.0,0.0,

YAXMAX....real vector used as an estimate of the maximum value of the ordinate for all curves in a given frame. This vector is used with YAXMIN in establishing an initial vertical axis range for each

frame.
YAXMAX=1.0,1.0,

ISPEC.....integer vector relating species numbers with curves. The vector ISPEC is used to identify the I^{th} curve with a graph of species ISPEC(I). This vector has IGNUM components. Frequently we select ISPEC(I)=I so that we simply plot each species. Note, however, that a given species can be associated with several curves, e.g. ISPEC(I)=I_0 for several values of I.
ISPEC=1,2,...,MXGNUM,

LGRØUP....integer vector used to associate a given curve with a given frame. In general, the I^{th} curve is plotted on the LGROUP(I) frame. Thus the I^{th} curve plots the ISPEC(I) species on the LGROUP(I) frame. Notice that if we wish to plot, say, the first species on frames one and two we use ISPEC(1)=ISPEC(2)=1, LGROUP(1)=1, and LGROUP(2)=2. There are IGNUM components in LGROUP.
LGROUP=1,1,..., ($1 \leq I \leq$ IGNUM)

LØRDER....integer vector used to order curves on frames. If IFORMT=1, the LORDER(I) calcomp plotting symbol is used for the I^{th} curve (see [10], p. 16) for a sample of these symbols). The selected symbol will be plotted at a few selected points along the I^{th} curve. If IFORMT$\neq$1, the separate axes plots in the frame will be ordered from the bottom of the frame, i.e. the I^{th} curve will be the LORDER(I) plot from the bottom of the LGROUP(I) frame. There are IGNUM components in LORDER. Usually we use LORDER(I)=I.
LORDER=1,1,..., ($1 \leq I \leq$ IGNUM)

A1.....real vector or r coordinates of the first endpoint of the cross-sections. There is a separate cross-section associated with each frame and thus there are max LGROUP(I) cross-sections to provide. Each cross-section is provided by giving the r and z components of the two endpoints of the cross-section line. Thus, for the I^{th} frame, the line used for the

cross-section connects (A1(I),B1(I)) with (A2(I),B2(I)).
A1=0.0,0.0,..., ($1 \leq I \leq$ MAX(LGROUP))

B1.....real vector of z coordinates of the first endpoint of the cross-sections.
B1=0.0,0.0,..., ($1 \leq I \leq$ MAX(LGROUP))

A2.....real vector of r coordinates of the second endpoint of the cross-sections.
A2=1.0,1.0,..., ($1 \leq I \leq$ MAX(LGROUP))

B2.....real vector of z coordinates of the second endpoint of the cross-sections.
B2=1.0,1.0,..., ($1 \leq I \leq$ MAX(LGROUP))

5.6 Namelist CNTRIN and Blackbox Input

In this subsection we describe the variables used in the contouring graphics package CØNTØR which appear in Namelist CNTRIN. This namelist, along with the variables in Namelist FØRMAT, are the input to a Mortran program which generates a dataset on unit 13 which in turn is input to the CØNTØUR.BLACKBØX package. Later in this section we discuss BLACKBØX control cards necessary for generating contour plots.

ISPEC.....integer vector of species numbers. The vector, just as the ISPEC vector in Namelist CSPIN, specifies that for a given time value, the Ith frame will be a contour plot of species ISPEC(I).
ISPEC=1,2,..., ($1 \leq I \leq$ IGNUM)

RMIN.....real vector of minimum r coordinate to be contoured on the Ith frame. There are IGNUM components in RMIN.
RMIN=0.0,0.0,..., ($1 \leq I \leq$ IGNUM)

RMAX.....real vector of maximum r coordinate to be contoured on the Ith frame. There are IGNUM components in RMAX.
RMAX=1.0,1.0,..., ($1 \leq I \leq$ IGNUM)

ZMIN.....real vector of minimum z coordinate to be contoured on the Ith frame. There are IGNUM components in ZMIN.
ZMIN=0.0,0.0,..., ($1 \leq I \leq$ IGNUM)

ZMAX.....real vector of maximum z coordinate to be contoured on the Ith frame. There are IGNUM components in ZMAX.
ZMAX=1.0,1.0,..., ($1 \leq I \leq$ IGNUM)

NRESR.....integer variable for the number of grid points used for the r direction. This variable defaults to the value of NRESIN in Namelist FØRMAT.
NRESR=NRESIN,

NRESZ.....integer variable for the number of grid points used for the z direction.
NRESZ=NRESIN,

After the execution of CØNTØR, a dataset is generated on unit 13 which is acceptable as input to CØNTØUR.BLACKBØX. This dataset consists of card images with a format of (1X,3F14.8,A3). While the information necessary to generate the BLACKBØX control cards is given in [11], we next give the specific values we used for these control cards in the examples. Notice that these are not JCL control cards, but rather control cards needed by BLACKBØX.

Card 1

1^TITLE

While a title can be given in columns 11-66, we left this blank.

Card 2

2^KEYS	1	1	1		13
(Column	20	24	32		49,50)

The only non-default values used are BØRDER (col. 20) to draw lines every inch on the plot, CØNTØUR LABELING (col. 24) to label every contour line, PØST DATA (col. 32) to plot all of the data points, and TAPE INPUT (cols. 49-50) to indicate that the data cards are on unit 13.

Card 3

3^SCALE			0.25		0.25
(Columns		31		40 41	50)

The only non-default values used are X UNITS PER INCH (cols. 31-40) and Y UNITS PER INCH (cols. 41-50). We set these so that we plot 0.25 units per inch in each direction.

Card 4

4^GRID		0.1	
(Columns	31		40)

The only non-default value used is CØNTØUR INTERVAL (cols. 31-40). This value is set to 0.1 so as to plot contours at intervals of 0.1. If this would cause more than 125 contours, BLACKBØX automatically resets the interval so that, at most, 125 contours are made.

Card 5

5^PØST

Card 6

6^FØRMAT		(1X,3E14.6,A3)	
(Columns	11		60)

The format of the data on unit 13 must be given in FØRTRAN specification for in columns 11-60.

Cards 1-6 are repeated as many times as there are contours to be plotted. For example, if we had IGNUM=1 and five time values there would be five contour plots to generate. In this case we would have five packets of Cards 1-6. After these cards we need a final card,

Card 7

8^STØP

5.7 Namelist DIM3IN

In this subsection we describe the values in Namelist DIM3IN which, together with those in Namelist FORMAT, provide the input data for the three-dimensional perspective program (THREED).

The variables ISPEC, RMIN, RMAX, ZMIN, ZMAX, NRESR, and NRESZ as described in section 5.6 are required to specify the species indices and the rectangular plotting domain in r-z space. These variables are not repeated here for the sake of brevity. We also need to specify the viewpoint of the surface and bounds on the function axis.

RVIEW.....real variable describing the r coordinate of the viewpoint. In terms of actual coordinates we view the three-dimensional surface from the point (RVIEW,ZVIEW,FVIEW). It is recommended that the viewpoint be initially selected to be a considerable distance from the surface. On subsequent runs the viewpoint may be refined. Note when ITIME=T, RVIEW is associated with the time axis.
RVIEW=100.0,

ZVIEW.....real variable specifying z coordinate of viewpoint.
ZVIEW=100.0,

FVIEW.....real variable specifying function axis coordinate of viewpoint.
FVIEW=100.0,

FMATMN....real variable used to provide an estimate of the minimum function value for all species over all time values. This variable, along with FMATMX, can be set large enough to provide a constant function axis for all plots in a given run. Otherwise, the code will adjust both FMATMN and FMATMX to accommodate the data.
FMATMN=0.0,

FMATMX....real variables used to provide an estimate of the maximum function value for all species over all time values.
FMATMX=1.0,

5.8 Summary of Restart Procedures

Recall that the program writes restart data on unit 11, and reads restart data from unit 10. Furthermore, the write on unit 11 will always occur when any of the following circumstances occur.

(1) If time expires during a steady-state calculation.

(2) At the conclusion of a steady-state calculation.

(3) If time expires during a transient calculation.

(4) At the conclusion of a transient calculation. Note that the value of NUTOUT determines where the transient calculation stops. For example, if the UTOUT array has 21 time values and we set NUTOUT=8, then the program will stop at the end of the first 8 time values in UTOUT. If one wished to continue, then on a subsequent restart, one would set $8 < \text{NUTOUT} \leq 21$. The program will then proceed from UTOUT(8) to UTOUT(NUTOUT).

The general procedure for a restart is as follows.

(a) Assign the previously written restart dataset to unit 10 (it was previously assigned to unit 11).

(b) Assign to unit 11 the dataset on which the restart data will be written at the conclusion of this run.

(c) In Namelist GRID set

```
INITSW=F,
GUESSW=F,
```

(d) If this is a restart from a steady-state calculation and is a continuation of a steady-state calculation, then leave STEDSW=T, and leave TRANSW as it was in the previous calculation. Then set ISTDRS=T,

(e) If the previous run was a normal conclusion of a steady-state calculation with DUMPSW=T, and if one wishes to restart from this run to do a transient calculation, set the following indicators: STEDSW=F, TRANSW=T, DUMPSW=T, (in this case DUMPSW=T, serves to indicate that transient will read the steady-state data).

(f) If the previous run was a transient calculation, and this is a restart to continue this calculation then set ITRARS=T, and make sure STEDSW=F,.

5.9 Job Control Language Usage

In this subsection the use of the DISPL code in terms of the IBM Job Control Language (JCL) used at Argonne is illustrated. The JCL provided merely illustrates one way of using the package, and we do not intend to limit the use of alternative approaches. Recall that DISPL consists of:

1. A file containing the Mortran Macros.
2. The computational phase of DISPL.
3. The cross-section plot graphics program.
4. The contour program.
5. The three-dimensional perspective program.

It is assumed that the user has copied these files into the following datasets:

```
B99999.MØRTM.DATA
B99999.DISPL.FØRT
B99999.CSP.FØRT
B99999.CØNTØR.FØRT
B99999.THREED.FØRT
```

and that the user has created space for the following object modules:

```
B99999.DISPL.ØBJ
B99999.CSP.ØBJ
B99999.CØNTØR.ØBJ
B99999.THREED.ØBJ
```

Finally, we assume that the user has created datasets to associate with unit 10 (for restart reading)

```
B99999.READ.DATA
```

unit 11 (for dump writing)

```
B99999.WRITE.DATA
```

and unit 12 (for graphics writing)

```
B99999.GRAPH.DATA
```

In order to use the computationed phase of DISPL, an object module of B99999.DISPL.FØRT is created. The following JCL will generate a FORTRAN H extended object module using cataloged procedures available at Argonne National Laboratory. For users outside of Argonne, Table 5.9.1 - 5.9.4 gives the actual cataloged procedures used here and later in this subsection.

```
//DISPL JØB (F99999,30,0,5),REGIØN=450K
Computer account card
//*MAIN ØRG=LØCAL
//STEP1 EXEC MØRC,CØMPLETE=IFEAAB,ØPTIØNS='XREF,ØPT=2'
//MØR.FT02F001 DD DSN=B99999.MØRTM.DATA,DISP=SHR
//MØR.SYSIN DD DSN=B99999.DISPL.FØRT,DISP=SHR
//FØR.SYSLIN DD DSN=B99999.DISPL.ØBJ,DISP=ØLD
/*
/* END ØF FILE
```

We are now ready to execute the computational phase of DISPL. It is assumed that the user has developed a card deck containing the user-supplied routines including MASTER DRIVE and a data deck containing the namelist cards.

```
//RUN JØB (F99999,10,0,5),REGIØN=500K,PRTY=N,
// CLASS=B
Computer account card
//*MAIN ØRG=LØCAL
//STEP1 EXEC MØRCLG,CØMPILE=IFEAAB,
// ØPTIØNS='XREF,ØPT=2'
//MØR.FT02F001 DD DSN=B99999.MØRTM.DATA,DISP=SHR
//MØR.SYSIN DD *

{User's card deck}

/*
//GØ.SYSIN DD DSN=B99999.DISPL.ØBJ,DISP=SHR
//GØ.FT10F001 DD DSN=B99999.READ.DATA,DISP=SHR
//GØ.FT11F001 DD DSN=B99999.WRITE.DATA,DISP=ØLD
//GØ.FT12F001 DD DSN=B99999.GRAPH.DATA,DISP=ØLD
//GØ.SYSIN DD *

{User's data deck}

/*
/* END ØF FILE
```

Notice that the MASTER DRIVER and all of the user-supplied routines are compiled in FØRTRAN H extended, assuming the original code is in MØRTRAN. If the user prefers, these routines can be written in FØRTRAN. In this case replace the STEP1 card and the three cards following it with

```
//STEP1 EXEC FTXCLG,ØPTIØNS='XREF,ØPT=2'
//FTX.SYSIN DD *
```

If there is no need for the graphics programs, no further JCL is required. If, however, graphics output is desired, the corresponding graphics package must be compiled. In the following, the JCL for generating object modules for each of the graphics programs is given. First, for CSP use

```
//CSP JØB (F99999,5,0,2),REGIØN=250K
Computer account card
//*MAIN ØRG=LØCAL
//STEP1 EXEC MØRC,CØMPILE=IFEAAB,ØPTIØNS='XREF,ØPT=2'
//MØR.FT02F001 DD DSN=B99999.MØRTM.DATA,DISP=SHR
//MØR.SYSIN DD DSN=B99999.CSP.FØRT,DISP=SHR
//MØR.SYSLIN DD DSN=B99999.CSP.ØBJ,DISP=ØLD
/*
/* END ØF FILE
```

To create the object module for CØNTØR, change the symbol CSP in the above JØB card and in the last two cards to CØNTØR. Similarly, to create the object module for THREED, change the symbol CSP in the JØB card and in the last two cards to THREED.

In using any of the graphics programs we assume that a dataset, B99999.GRAPH.DATA, has been written during an execution of the computational phase of DISPL. This is accomplished by assigning it to unit 12 and setting GRAPH=T in Namelist DATA. The JCL given below generates spooled STATØS graphics output at Argonne. The JCL for alternative graphics devices at Argonne is discussed in the Graphics chapter of the ANL User's Guide [15]. At installations other than Argonne, the user would need to verify that he had the corresponding graphics package (CALCØMP, CØNTØUR.BLACKBØX, or DISSPLA) as well as make the appropriate changes in the JCL.

To execute the CSP program, use

```
//CSPEX JØB (F99999,1,0,1),REGIØN=250K
Computer account card
//*MAIN ØRG=LØCAL
//STEP2 EXEC FTXCLG,PRELIB='SYS1.CALLIB'
//FTX.SYSIN DD *
{Single precision version of
{SUBRØUTINE ANAL or a dummy version}
/*
//GØ.SYSLIN DD DSN=B99999.CSP.ØBJ,DISP=SHR
//GØ.LDRIN DD DSN=SYS1.PLØTTER.SPSTATØS(PLØTTER),DISP=SHR
//GØ.FT12F001 DD DSN=B99999.GRAPH.DATA,DISP=SHR
//GØ.GRAPHICS DD SYSØUT=S
//GØ.SYSIN DD *
{Data cards}
/*
/* END ØF FILE
```

To execute the CØNTØR program, use the following JCL.

```
//CØNTE JØB (F99999,1,0,1),REGIØN=350K
Computer account card
//*MAIN ØRG=LØCAL
//STEP2 EXEC FTXLG
//GØ.SYSLIN DD DSN=B99999.CØNTØR.ØBJ,DISP=SHR
//GØ.FT12F001 DD DSN=B99999.GRAPH.DATA,DISP=SHR
//GØ.FT13F001 DD DSN=&&TMP1,UNIT=SASCR,SPACE=(6000,(20,20)),
// DISP=(NEW,PASS)
//GØ.SYSIN DD *
{Data cards}
/*
//STEP3 EXEC PGM=G1DATA
//STEPLIB DD DSN=SYS1.PLØTPKG.CØNTØUR.BLACKBØX,DISP=SHR
//FT05F001 DD DDNAME=CARDIN
//FT06F001 DD SYSØUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1463)
//FT09F001 DD UNIT=(SASCR),SPACE=(CYL,(2,1))
//FT10F001 DD UNIT=(SASCR),SPACE=(CYL,(2,1))
//FT13F001 DD DSN=&&TMP1,DISP=(ØLD,DELETE)
//GØ.CARDIN DD *
{CØNTØUR.BLACKBØX Control cards}
/*
//GØ.GRAPHICS DD DSN=&&TMP2,UNIT=SASCR,SPACE=(6000,(20,20)),
// DISP=(NEW,PASS)
//STEP4 EXEC PGM=SPSTATØS
//STEPLIB DD DSN=SYS1.PØSTPLØT.G1DATA,DISP=SHR
//GRAPHIN DD DSN=&&TMP2,DISP=(ØLD,DELETE)
//GRAPHICS DD SYSØUT=S
//FT06F001 DD SYSØUT=A
/*
/* END ØF FILE
```

To execute the THREED program, use the above JCL for the CSP execution with the following changes.

(a) Replace the JØB card with

```
//D1M3EX JØB (F99999,1,0,1),REGIØN=250K
```

(b) Replace the EXEC card with

```
//STEP2 EXEC FTXLG,PRELIB='SYS1.D1SLIB'
```

(c) Remove FTX.SYSIN card, the code for SUBRØUTINE ANAL, and the /* card.

(d) Replace the SYSLIN card with

```
//GØ.SYSLIN DD DSN=B99999.THREED.ØBJ,DISP=SHR
```

(e) Use the data cards for THREED.

Any of the above graphics executions can be combined with the computational phase execution into a single job. To illustrate the procedure consider including the CSP execution in the computational phase execution, i.e. in the JØB named RUN. Just before the END ØF FILE card, insert the JCL for the CSPEX JØB beginning with the STEP2 card. Two changes in the graphics JCL need to be made:

(a) The STEP2 card is changed to

```
//STEP2 EXEC FTXCLG,PRELIB='SYS1.CALLIB',CØND=(13,LE,STEP1.GØ)
```

This change prohibits the execution of the graphics step if the condition code at the conclusion of the computational phase is 13 or more. The computational phase will set such a condition code if there is an abnormal termination, e.g. failure in GEAR.

(b) The GØ.FT12FOOI card is changed to

```
//GØ.FT12FOOI DD DSN=*.STEP1.GØ.FT12FOOI,DISP=SHR
```

This card uses a referback to indicate that the graphics dataset is the same one used in the GØ.FT12FOOI card in the computational phase. The advantage of using referbacks is that the name of the graphics dataset occurs only once in the JCL. Thus, if the user has a number of graphics datasets for various problems, he need change only one card (the GØ.FT12FOOI in the computational phase) to indicate the appropriate dataset.

For users outside of Argonne we provide the cataloged procedures in Tables 5.9.1-5.9.4 which have been used in the examples of JCL in this section. Further, the DISPL computational code uses a clock function, TLEFT, in subroutines EXEC, STEADY, and TIMEX to monitor computer time. This function, which is available on Argonne's IBM 370/195 has the format

```
TLEFT(DUM)
```

where DUM is a single precision real dummy variable. The function value returned is the time left for the job in centiseconds. Users wishing to install DISPL on another system should provide an interface routine to fit this format or use a dummy routine to return a large constant value. Also, subroutine EXEC sets condition codes of 13 or greater if the run is abnormal. This feature is available principally to inhibit the graphics step as outlined above. If the

user does not wish to set condition codes, subroutine EXEC should be modified. The rest of DISPL is written using ANSI Fortran functions and no local Argonne packages (except for the graphics CALCØMP, CØNTØUR.BLACK, or DISSPLA software and graphics cataloged procedures).

Table 5.9.1. MØRC Catalogued Procedure

```
MEMBER NAME  MORC
//MORC     PROC REGN=250K,OPTIONS=,COMPILE=IGIFORT,
//  OBJUNIT=SASCR,OBJECT='&OBJT',
//  OBJROOM='(3120,(40,10),RLSE)',OBJBLK=3120
//MCR      EXEC PGM=MORTRAN,REGION=&REGN
//STEPLIB  DD   DISP=SHR,DSN=B23531.MORTRAN.LOAD
//FT06F001 DD SYSOUT=A,DCB=(LRECL=133,BLKSIZE=1330,RECFM=FBA)
//FT01F001 DD   DISP=SHR,DSN=B23531.MORTMACS.DATA
//FT07F001 DD   DSN=&FINPUT,DISP=(,PASS),UNIT=SASCR,
//              SPACE=(TRK,(2,2)),DCB=(LRECL=80,BLKSIZE=1680,RECFM=FB)
//FT05F001 DD   DDNAME=SYSIN
//FOR  EXEC  PGM=&COMPILE,REGION=&REGN,PARM='NOTERM,&OPTIONS'
//SYSLIN   DD  DDNAME=SYSOBJ
//SYSOBJ   DD  UNIT=&OBJUNIT,DSNAME=&OBJECT,SPACE=&OBJROOM,
//  DISP=(MOD,PASS),DCB=(RECFM=FB,LRECL=80,BLKSIZE=&OBJBLK)
//SYSPRINT DD  SYSOUT=A,DCB=(RECFM=FBA,LRECL=120,BLKSIZE=1560)
//SYSPUNCH DD  SYSOUT=B,DCB=(RECFM=F,BLKSIZE=80)
//SYSIN    DD DSN=&FINPUT,DISP=OLD
//* A MORTRAN PREPROCESSOR PROCEDURE SUPPORTED BY
//* CLARK PEWITT, AMD     MAY  21,1975
```

Table 5.9.2. MØRCLG Catalogued Procedure

```
MEMBER NAME  MORCLG
//MORCLG   PROC REGN=250K,OPTIONS=,COMPILE=IG1FORT,
//  OBJUNIT=SASCR,OBJECT='&OBJT',
//  OBJROOM='(3120,(40,10),RLSE)',OBJBLK=3120,
//  LPARM=,EP=,GOPARM=,
//  PRELIB='SYS1.DUMMYLIB',POSTLIB='SYS1.DUMMYLIB',
//  AMDLIB='SYS1.AMDLIB2',LIBRARY='SYS1.FORTLIB2',
//  OBJDISP='(OLD,DELETE)',
//  GOIF='(5,LT,FOR)',GOREGN=128K
//MOR      EXEC PGM=MORTRAN,REGION=&REGN
//STEPLIB  DD   DISP=SHR,DSN=B23531.MORTRAN.LOAD
//FT06F001 DD SYSOUT=A,DCB=(LRECL=133,BLKSIZE=1330,RECFM=FBA)
//FT01F001 DD   DISP=SHR,DSN=B23531.MORTMACS.DATA
//FT07F001 DD   DSN=&FINPUT,DISP=(,PASS),UNIT=SASCR,
//              SPACE=(TRK,(2,2)),DCB=(LRECL=80,BLKSIZE=1680,RECFM=FB)
//FT05F001 DD   DDNAME=SYSIN
//FOR  EXEC  PGM=&COMPILE,REGION=&REGN,PARM='NOTERM,&OPTIONS'
//SYSLIN   DD  UNIT=&OBJUNIT,DSNAME=&OBJECT,SPACE=&OBJROOM,
//  DISP=(MOD,PASS),DCB=(RECFM=FB,LRECL=80,BLKSIZE=&OBJBLK)
//SYSPRINT DD  SYSOUT=A,DCB=(RECFM=FBA,LRECL=120,BLKSIZE=1560)
//SYSPUNCH DD  SYSOUT=B,DCB=(RECFM=F,BLKSIZE=80)
//SYSIN DD DSN=&FINPUT,DISP=OLD
//GO     EXEC  PGM=LOADER,PARM='MAP,PRINT,&LPARM,EP=&EP/&GOPARM',
//             COND=&GOIF,REGION=&GOREGN
//FT05F001 DD  DDNAME=SYSIN
//FT06F001 DD  SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1596)
//FT07F001 DD  SYSOUT=B
//SYSLIB   DD  DISP=SHR,DSN=&PRELIB
//         DD  DISP=SHR,DSN=&AMDLIB
//         DD  DISP=SHR,DSN=&LIBRARY
//         DD  DISP=SHR,DSN=&POSTLIB
//SYSLIN   DD  DSNAME=*.FOR.SYSLIN,DISP=&OBJDISP
//         DD  DDNAME=LDRIN
//SYSLOUT  DD  SYSOUT=A,DCB=(LRECL=121,BLKSIZE=1573)
//* A MORTRAN PREPROCESSOR PROCEDURE SUPPORTED BY
//* CLARK PEWITT, AMD      MAY 21, 1975
```

Table 5.9.3. FTXCLG Catalogued Procedure

```
MEMBER NAME  FTXCLG
//FTXCLG   PROC REGN=240K,OPTIONS=,COMPILE=IFEAAB,
//  OBJUNIT=SASCR,OBJECT='&OBJT',
//  OBJROOM='(CYL,4)',OBJBLK=3120,
//  LPARM=,EP=,GOPARM=,
//  PRELIB='&DUMMYLIB',POSTLIB='&DUMMYLIB',
//  AMDLIB='&AMDLIB2',LIBRARY='&FORTLIB2',
//  OBJDISP='(OLD,DELETE)',
//  GOIF='(5,LT,FTX)',GOREGN=128K
//FTX  EXEC  PGM=&COMPILE,REGION=&REGN,PARM='NOTERM,&OPTIONS'
//SYSLIN   DD  UNIT=&OBJUNIT,DSNAME=&OBJECT,SPACE=&OBJROOM,
//  DISP=(MOD,PASS),DCB=(RECFM=FB,LRECL=80,BLKSIZE=&OBJBLK)
//SYSPRINT DD  SYSOUT=A,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=1511)
//SYSPUNCH DD  SYSOUT=B,DCB=(RECFM=F,BLKSIZE=80)
//SYSUT1   DD  SPACE=(TRK,(0,19)),UNIT=(SASCR)
//SYSUT2   DD  SPACE=(TRK,(0,19)),UNIT=(SASCR,SEP=(SYSUT1))
//GO     EXEC  PGM=LOADER,PARM='MAP,PRINT,&LPARM,EP=&EP/&GOPARM',
//             COND=&GOIF,REGION=&GOREGN
//FT05F001 DD  DDNAME=SYSIN
//FT06F001 DD  SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1596)
//FT07F001 DD  SYSOUT=B
//SYSLIB   DD  DISP=SHR,DSN=&PRELIB
//          DD  DISP=SHR,DSN=&AMDLIB
//          DD  DISP=SHR,DSN=&LIBRARY
//          DD  DISP=SHR,DSN=&POSTLIB
//SYSLIN   DD  DSNAME=*.FTX.SYSLIN,DISP=&OBJDISP
//          DD  DDNAME=LDRIN
//SYSLOUT  DD  SYSOUT=A,DCB=(LRECL=121,BLKSIZE=1573)
//*  THIS PROCEDURE IS DOCUMENTED IN CHAPTER 9 OF THE USER'S GUIDE.
```

Table 5.9.4. FTXLG Catalogued Procedure

```
MEMBER NAME  FTXLG
//FTXLG    PROC  LPARM=,EP=,GOPARM=,
//  PRELIB='&DUMMYLIB',POSTLIB='&DUMMYLIB',
//  AMDLIB='&AMDLIB2',LIBRARY='&FORTLIB2',
//  GOREGN=128K
//GO     EXEC  PGM=LOADER,PARM='MAP,PRINT,&LPARM,EP=&EP/&GOPARM',
//             REGION=&GOREGN
//FT05F001 DD  DDNAME=SYSIN
//FT06F001 DD  SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1596)
//FT07F001 DD  SYSOUT=B
//SYSLIB   DD  DISP=SHR,DSN=&PRELIB
//          DD  DISP=SHR,DSN=&AMDLIB
//          DD  DISP=SHR,DSN=&LIBRARY
//          DD  DISP=SHR,DSN=&POSTLIB
//SYSLIN   DD  DDNAME=LDRIN
//SYSLOUT  DD  SYSOUT=A,DCB=(LRECL=121,BLKSIZE=1573)
//*  THIS PROCEDURE IS DOCUMENTED IN CHAPTER 9 OF THE USER'S GUIDE.
```

5.10 Error Messages

This program performs some consistency checks on the input stream. We distinguish between fatal and non-fatal errors in the input. Whenever possible an error is interpreted as non-fatal in which case the error is corrected and execution resumes. When a non-fatal error is encountered, the error is corrected and a message is printed informing the user of the correction. For example, if the number of quadrature points per interval in the r direction NQR is less than KR-1, then the quadrature error may exceed the spatial truncation error, and even worse, the matrix AL can be so poorly conditioned that it cannot be inverted. Hence if NQR < KR-1, the code will set NQR = KR-1 and then continue. This same situation also applies to NQZ.

For fatal errors, the program prints a message and then stops. The following errors are recognized as fatal errors in the computational phase of DISPL.

(1) If side J has a non-essential boundary condition for the K-th species, we have

$$\alpha hu+\beta\mathcal{D}\nabla u\cdot n = \gamma h\rho^0$$

on side J where $\beta = \beta(K,J) \neq 0$. Thus the code checks that NSJ(K) and $\beta(K,J)$ are not both zero. If they are, the following message appears.

***INPUT ERRØR.BETA=0. FOR A NØN-ESS. SIDE.

(2) Recall that MAXBRK is the Macro parameter for the maximum number of intervals in either direction. The code checks whether LR > MAXBRK-1 or LZ > MAXBRK-1, and if so prints

***INPUT ERRØR.NUMBER ØF INTERVALS NØT CØNSISTENT WITH
NUMBER ØF BREAKPØINTS.

(3) NRNZ is the number of variables associated with one species, and MXNRNZ is the Macro parameter for this variable. The code checks NRNZ > MXNRNZ, and if so prints

***INPUT ERRØR.NUMBER ØF VARIABLES PER SPECIES EXCEEDS
DIMENSIØNED SIZE.

(4) NVAR is the total number of variables for the problem, and MXNVAR is the corresponding Macro parameter. The code checks NVAR > MXNVAR and if so prints

***INPUT ERRØR.TØTAL NUMBER ØF VARIABLES EXCEEDS
DIMENSIØNED SIZE.

(5) NQR(NQZ) is the number of quadrature points per interval, and MAXTQD is the corresponding Macro parameter. The code checks NQR > MAXTQD or NQZ > MAXTQD and if so prints

***INPUT ERRØR.NUMBER ØF QUADRATURE PØINTS EXCEEDS
DIMENSIØNED SIZE.

(6) NUTØUT is the number of output times in a transient calculation. The code checks if NUTØUT < 2 or NUTØUT > 100, and if so prints

***INPUT ERRØR.NUTØUT IS ØUTSIDE THE INTERVAL (2,100).

(7) The PW array is dimensioned at run time by the user setting NPW in the DRIVER. The code checks whether the user has set NPW large enough, that is, whether MFBW*NVAR > NPW for the direct version, or FBW*NVAR*NSPEC > NPW for the iterative version. In either case the code prints

***PW ARRAY IS TØØ SMALL.

(8) The AL array is dimensioned at run time by setting NAL in MASTER DRIVER. For either version the code checks whether FBW*NRNZ > NAL, and if so prints

***AL ARRAY IS TØØ SMALL.

(9) MAXNØT is the Macro variable for the number of knots in either the r or z direction. If this number is exceeded the code prints

***INPUT ERRØR. TØTAL NUMBER ØF KNØTS IN
ØNE DIRECTIØN EXCEEDS STØRAGE.

The graphics programs generate fatal errors with messages if the Macro variables used in the graphics programs are not as large as the Macro variables used to generate the graphics dataset on unit 12 during the execution of the computational phase. In the following discussion we refer to the value of the Macros used in the graphics codes by preceding the variable name with a J. The corresponding value in the computational code will be indicated by preceding the variable name with an I.

The fatal errors are:

(1) If the original Macro for the number of breakpoints is greater than the corresponding graphics Macro, the message

SPECIFIED NUMBER ØF BREAKPØINTS (IMAXBRK) EXCEEDS THE
MAXIMUM NUMBER DIMENSIØNED (JMAXBRK). RUN STØPPED.

(2) If the original Macro for the number of species is greater than the corresponding graphics Macro the following message appears.

SPECIFIED NUMBER ØF SPECIES (IMAXSP) EXCEEDS THE
MAXIMUM NUMBER DIMENSIØNED (JMAXSP). RUN STØPPED.

(3) If the original Macro for the spline order is greater than the value used in the graphics program the following message appears.

SPECIFIED ØRDER ØF SPLINES (IMAXK) EXCEEDS THE
LIMITS ØF THE PRØGRAM (JMAXK). RUN STØPPED.

(4) If the original Macro for the number of knots in either direction is greater than the value used in the graphics program the following message appears.

SPECIFIED NUMBER ØF KNØTS (IMAXNØT) EXCEEDS THE
MAXIMUM NUMBER DIMENSIØNED (JMAXNØT). RUN STØPPED.

(5) If the original Macro for the number of variables in either direction is greater than the value used in the graphics program the following message appears.

THE SINGLE SPECIES NUMBER ØF VARIABLES (IMXNRNZ) EXCEEDS
THE LIMITS ØF THE PRØGRAM (JMXNRNZ). RUN STØPPED.

(6) If the original Macro for the total number of variables is greater than the value used in the graphics program the following message appears.

THE SPECIFIED NUMBER ØF VARIABLES (IMXNVAR) EXCEEDS
THE LIMITS ØF THE PRØGRAM (JMXNVAR). RUN STØPPED.

In addition, a fatal error can occur when ITIME=T and if NTIME is greater than NRES1 (for CSP) or NRESD (for CØNTØR or THREED). If this occurs, the following message is printed.

SPECIFIED NUMBER ØF TIME VALUES IS
GREATER THAN $\begin{Bmatrix} \text{NRES1} \\ \text{NRESD} \end{Bmatrix}$, NTIME= . RUN STØPPED.

where the appropriate Macro name NRES1 or NRESD is printed as well as the value of NTIME.

If a fatal error is detected (either due to input or due to computation), the last message printed in EXEC is

EXIT EXEC WITH ABNØRMAL TERMINATIØN

and control is returned to the Master Driver.

The following non-fatal message is printed by the graphics programs is ITIME=T and there are more time values on the graphics dataset than the user requested to be graphed via NTIME.

WARNING *** MØRE TIME VALUES THAN NTIME.

6. DESCRIPTION OF PRINTED AND GRAPHICAL OUTPUT

In this section we discuss the output generated by the DISPL package. We first discuss the printed output generated by the computational phase of DISPL in section 6.1. In section 6.2 we discuss the printed and graphical output which can be produced by the cross-section plot program. In section 6.3 we discuss the printed and graphical output which can be produced by the contour package. Finally in section 6.4 we discuss printed and graphical output which can be produced by the three-dimensional perspective program. In all cases, we describe the output with reference to the examples in section 7.

6.1 Computational Printed Output

In this subsection we discuss the printed output of the DISPL computational code. The printed output has three main sections. The first section is the result of processing Namelist GRID and the first read of Namelist DATA. The second section contains the results of any initial fit computation and any steady-state calculation. The third section contains the results of a second read of Namelist DATA and any transient calculation which may have occurred. In discussing the results of processing the namelists, we will discuss only those aspects which are not clear from the discussion of the namelists in section 5.

Considering the first section, we use example 7.2 as an illustration. The output begins with a summary of the variables in Namelist GRID. Notice that INITIAL CØNTR=13; this is just an indication that the code is going to set CØNTR=KR-1 which is the default value. The next part of the output consists of certain internal arrays followed by a summary of Namelist DATA. The code first indicates that CØNTR is reset to KR-1=4-1=3 and CØNTZ is reset to KZ-1=1-1=0. NVGAP is the number of vertical gap interfaces, and NHGAP is the number of horizontal gap interfaces. LR and LZ are the number of intervals in the r and z direction. The IL array is the index set for the B-splines in the r-direction. JL is the corresponding index set for the z direction. IREF is an index set in the r direction for the interfaces. JREF is a similar array for the z direction.

MLTAB is the two-dimensional table of materials in the domain. This table is the internal array which is obtained from the Namelist GRID array

MATL. Next, the code determines whether it is more efficient (in terms of bandwidth) to store the coefficients with horizontal or vertical ordering. The ordering used is reported in the output. The next two lines of code deal with constants involved in the indexing of the coefficients. These constants are explained in section 4.10.

The output next gives the actual system size for the problem. NR is the number of variables per species in the r-direction. NZ is the corresponding number for the z-direction. NVAR is the total number of variables, i.e. NSPEC*NR*NZ. The output next indicates whether this is the iterative or direct version of the code. The summary print of namelist DATA then begins. The reference to PRØUT in describing the time and space grid is to SUBRØUTINE PRØUT which controls output (both for units 6 and 12) during the steady-state and transient computation. The variable MF (which is 21) is an indicator for the ODE solver GEAR and must remain fixed.

Next we consider the second section. After completing the summary print for Namelist DATA, the code calls SUBRØUTINE INIFIT if INITSW=T (which is the case in this example). The output contains a table of the least-squares solution evaluated at the quadrature points. In this example the code next reads namelist DATA for the second time and repeats its summary print. Notice that the coefficients, W, have changed as a result of the call to INIFIT. If GUESSW and/or STEDSW were TRUE, calls to SUBRØUTINE GUESS1 and/or STEADY would have occurred. While the call to GUESS1 would not directly produce any printed output, the call to STEADY would produce output similar to the transient output discussed below. Such steady-state output would occur after the first time-step, after every ISTDFQ steps beyond the first one, and at the final step. Also, if the code runs out of time during the execution of STEADY, printout will occur for the last step before doing a dump.

Finally, we consider the third section of output. We first see a message indicating that the transient solution is beginning. There then follows a set of small (half page) reports of the solution as it is progressing. Each report is a result of a call to PRØUT and occurs at a time specified by the user's time grid. Consider the first report in example (section 7.1) in detail. The time value is given followed by the spline coefficient values. Each approximate solution is then evaluated on the user-specified grid and, if IANAL=T, the corresponding analytic solution is given. The output values on

this grid are arranged as follows: each row represents a fixed value of z with the first row corresponding to ZGRID(JZGRD) and the last row corresponding to ZGRID(1). On a given row, the first entry corresponds to RGRID(1) and the last entry corresponds to RGRID(IRGRD).† The message

(FRØM PRØUT VIA TIMEX)

is a reminder that this is a transient solution. If this call to PRØUT was from SUBRØUTINE STEADY we would get the following message.

(FRØM PRØUT VIA STEADY)

After the final species solution is printed, the time value, TØUT, is again printed.

PRØUT reports other than the first one also include the value DELTA T and provide information concerning the monitoring of computer time usage.

DELTA T is the current spacing of subintervals on the user's time grid. Notice that this number will change when we move from one major time interval to another, i.e. from [UTØUT(1),UTØUT(2)] to [UTØUT(2),UTØUT(3)]. In the case of a steady-state solution, DELTA T is the actual time step being used in the ODE solver.

The next two lines deal with monitoring computer time usage. Based on the amount of time required to solve the problem during the previous call to the ODE solver GEAR, the code estimates the time required to do the next call to GEAR. This estimate is given, followed by the amount of computer time remaining. If the estimate is less than the time left we proceed. If not, we immediately dump the current results on unit 11, report this fact on the output, and stop. In the steady-state case a similar situation applies.

Returning to the example in section 7.1, after the transient solution is completed, the code writes the current solution on unit 11 and reports this on the output as a normal dump. The value of I (=5 in section 7.1) is the final index of UTØUT used. Finally, the code checks whether there is another problem to solve. If not, a message indicating that the end of the input file has been reached is printed.

---

†It should be emphasized that the output from PRØUT (whether from STEADY or TIMEX is laid out just as if one were looking at the domain. That is, the output consists of a series of rows of numbers. Each row has values corresponding to ascending values of RGRID. The first

6.2 Cross Section Plotting

The output of the cross section plotting program (CSP) is both printed and graphical. In this subsection we discuss both forms of output by reference to the examples in section 7. To begin with, all of the graphing programs can be used in one of two basic modes. If ITIME=F in Namelist FORMAT, the graphing program will generate plots associated with the solution at a specific time value. For example, CSP will generate plots of species solutions along one-dimensional slices at time values associated with the time array specified in Namelist DATA. If, on the other hand, ITIME=T, the graphing program will generate graphs with time as one of the independent variables. In the case of CSP we will obtain plots as specified by the graphing formats in Namelists FORMAT and CSPIN. The plot involves a graph of a species value $u_k(r^*,z^*,t)$, at a fixed spatial location (r^*,z^*) as a function of time t for a given species k. The point (r^*,z^*) is specified as follows. Recall that for the m^{th} frame, CSP requires that a line segment in the spatial domain be specified by giving its end points (A1(m),B1(m)) and (A2(m),B2(m)). The point (r^*,z^*) is specified by setting

$$A1(m)=r^*,\ A2(m)=r^*,$$
$$B1(m)=z^*,\ B2(m)=z^*,$$

Note that $r^* = r^*(m)$, $z^* = z^*(m)$, i.e. (r^*,z^*) can change with each frame. The time values used in these plots are specified through the users time grid in Namelist DATA. The ITIME=T case for CØNTØR and THREED will be discussed in sections 6.3 and 6.4.

We now discuss the printed output of CSP when using the ITIME=F mode. For example, consider the output in section 7.2. The printed output begins with:

BEGINNING CROSS SECTION GRAPHICS PACKAGE

The program then checks the sizes of various macro variables. If these values are acceptable, no prints occur. The program then gives formatting parameters and the value of NRESIN. The values of ITIME and NTIME are then given. Recall that NTIME is used only if ITIME=T. The values of YAXMIN(I) and YAXMAX(I) are then given in pairs for I=1,...,IGNUM. Next, the user's graphing format data is summarized on a frame-by-frame basis. Specifically, for each frame (at a given time value) the ordering of plots on that frame

and the associated species number is given. In a similar manner. the definition of the cross section for each frame is reported. Finally, the program informs the user that the end of data has been encountered and the total number of frames produced is reported.

The graphical output associated with an ITIME=F case is also illustrated in section 7.2. In this example graphs were produced in the separate (not packed) format. Notice that the legend in the lower left corner of each graph gives the species number. Further, the time value associated with each frame is given. (Warning: Since floating point numbers must be plotted in an F format in the CALCOMP package, we sometimes obtain a double asterisk when a number, such as time, is out of range.)

We next consider the ITIME=T case, as is illustrated in section 7.3. The printed output is the same as in the ITIME=F case and for reasons of space we do not include it. The graphical output for ITIME=T differs from that for ITIME=F in that the legend in the lower left corner gives the r and z coordinates of the point where the solution is evaluated rather than a time value. Of course, the horizontal axis is time rather than r or z. The most significant difference is that the graph for ITIME=F is a curve whereas the graph for ITIME=T is just the NTIME points at which we have evaluated the solution. Finally, we point out that even when ITIME=T we can use the other options of CSP such as the grouping format (packed or separate) as well as the ISPEC, LGROUP, and LORDER arrays.

6.3 Contour Plotting

The contouring program (CØNTØR) can be used in one of two basic modes. If CØNTØR is used with ITIME=F contour plots are generated in which each plot is the approximate solution for a particular species over a rectangular domain in r-z space at a particular time. Thus we generate a series of contour plots ranging over the species and time values. If CONTR is used with ITIME=T, we generate a given contour plot with two independent variables -- time and a space variable. The time values used are those defined in Namelist DATA and must be uniformly spaced for use in CØNTØR. The spatial variable must be a line in the r-z domain and must be parallel to one of the coordinate axes. The definition of this line is transmitted to CØNTØR through the RMIN, RMAX, ZMIN, and ZMAX arrays (which define the contouring domain when ITIME=F). Notice that if the problem is one dimensional in the spatial variable, then a contour plot cannot be done with ITIME=F,; however a contour plot can be done with ITIME=T.

In describing the output of CØNTØR we consider the ITIME=F mode and refer to the examples in section 7 for concreteness. The CØNTØR program itself generates a data file on unit 13 which is in a format for use in CONTOUR.BLACKBOX. Thus, in order to generate a contour plot we require two job steps. The BLACKBOX control cards we use in plot generation are described in section 5.6. The printed output of the CØNTØR program is illustrated in section 7.1. The output begins with the statement

BEGINNING CONTOUR GRAPHICS PACKAGE

The program then does a series of checks on the sizes of Macro variables. If these values are within range, the execution continues. The output next states the VERSION of CØNTØR being used followed by the value of IGNUM. The user's formatting instructions are then summarized. Next, the value of ITIME is given along with the number of points in the r and z directions (NRESR and NRESZ) each of which is the input value NRESIN. Then, as each time value is being processed, the time is given followed by the labels R, Z, and F.

The printed output of the second job step (CONTOUR BLACKBOX) is also illustrated in section 7.1 where we have used the BLACKBOX control cards described in section 5.6. The printed output consists of a summary page associated with each contour plot. Note that these summary outputs could be

enhanced by using certain options on the control cards. For example, the units of the X Y SCALE default to feet per inch. Other units can be specified in columns 21-24 of the POST card in the BLACKBOX control card deck. There would, however, be no change in the graphical output.

The graphical output of the second job step is illustrated in section 7.1. The domain of the graph is subdivided into rectangles and the actual domain [RMIN(I),RMAX(I)] × [ZMIN(I),ZMAX(I)] consists of all the subrectangles except for the uppermost and rightmost ones. The datapoints generated by CONTR are indicated by either cross-hatches "+" or by "-" or "|" when the datapoints happen to intersect grid lines. The contours are indicated by curves with the corresponding function value.

6.4 Perspective Plotting of Solution Surfaces

The three-dimensional perspective program (THREED) uses the DISSPLA graphics software package to generate views of the solution surface for a particular species over a rectangular domain. As with CSP and CONTR, THREED can be used in two basic modes: ITIME=F and ITIME=T. When used with ITIME=F, the domain is specified by [RMIN(I),RMAX(I)] × [ZMIN(I),ZMAX(I)]. When used with ITIME=T the domain is defined as for CONTR. That is, the domain involves two independent variables: the first is time and values must be generated by a uniformly spaced time grid in Namelist DATA; the second independent variable must be parallel to the r or z axis and is specified by the values of RMIN(I), RMAX(I), ZMIN(I), and ZMAX(I). Notice that when ITIME=T, the spatial domain can, in fact, arise from a one-variable problem (see section 7.3).

In discussing the printed output we refer to the example in section 7.1. While this output deals with ITIME=F, the printed output for ITIME=T is essentially the same. The output begins with

BEGINNING THREE DIMENSIONAL GRAPHICS PACKAGE

The program then conducts a series of checks for the size of certain variables relative to the size of the macro variables in the particular compilation being used. If necessary, diagnostic messages are given and the program stops. In the normal case the program next provide a summary of the input data. First, the VERSION and value of IGNUM are given. This is followed by the

user's formatting instructions and the values of NRESIN and ITIME. Then the values of ISPEC, RMIN, RMAX, ZMIN, and ZMAX are given. Finally the viewpoint and FMATMN and FMATMX are given.

The program next reports

TIME FØR THIS RUN IS

and gives each value of TØUT being graphed followed by a DISSPLA plot summary.

The graphical output for THREED gives a series of frames which are perspective views. Each view is a surface for a given species at a given time. Notice that each frame gives the title, value of TØUT, frame number, and specie number. Thus, each frame is clearly defined without reference to the printed output.

When using THREED with ITIME=T the outputs are similar. The printed output differs in only one respect:

> The user's stated number of time values, NTIME (which is only required when ITIME=T) must not be greater than the Macro variable NRESD. A check of this requirement is made and, if necessary, a diagnostic message is printed. Further, if the actual number of time values is greater than NRESD, a warning message is given and the graph produced represents only the first NRESD time values.

The graphical output for the ITIME=T plots is similar to the graphical output already described. Of course, TØUT is not given in the lower left corner and one of the axes is the time axis.

7. SAMPLE PROBLEMS

In this section, we will illustrate the procedure used in setting up problems for this code.

7.1 Two-Dimensional Heat Conduction in a Plate

Consider a rectangular plate $R = \{(r,z): 0 \leq r \leq 2,\ 0 \leq z \leq 1\}$ with the heat conduction equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial r^2} + \frac{\partial^2 u}{\partial z^2} \text{ in } R \text{ for } t > 0 . \tag{7.1.1}$$

Let the following temperature distribution be specified on the sides for $t > 0$.

$$\begin{aligned} &u(0,z,t) = \phi(z,t), \\ &u(r,0,t) \equiv 0, \\ &u(2,z,t) \equiv 0, \\ &u(r,1,t) \equiv 0, \end{aligned} \tag{7.1.2}$$

and let the initial temperature distributions be

$$u(r,z,0) \equiv 0. \tag{7.1.3}$$

For $\phi(z,t)$ we take

$$\phi(z,t) = t\theta(z) \tag{7.1.4}$$

where

$$\theta(z) = \begin{cases} 2z \text{ for } 0 \leq z \leq \frac{1}{2} \\ 2(1-z) \text{ for } \frac{1}{2} \leq z \leq 1 \end{cases} \tag{7.1.5}$$

To set up this problem, we have to consider the three Namelists described in section 5, the user-supplied subroutines described in section 4, and the Namelist graphics data described in section 5.

Namelist GRID

1. Spline Order

   When in doubt, use a cubic, i.e.
   KR=4, KZ=4, (Default) (Default cases do not require any cards.)

2. Continuity at the mesh points

   With a homogeneous domain and smooth data, the smooth splines are a reasonable choice.
   CØNTR=3, CØNTZ=3, (= KR-1 and KZ-1) (Default)

3. Number of species
   NSPEC=1, (Default)

4. Domain
   RLØW=0.0, (Default)
   RUP=2.0,
   ZLØW=0.0, (Default)
   ZUP=1.0, (Default).

5. Total number of interfaces
   None (Default)

6. Interface mesh points
   None (Default)

7. Interface type codes
   None (Default)

8. Geometry type
   Rectangular
   DELTA=0, (Default)

9. Total number of non-interface mesh points in each direction. (This total does not include the end points.)

   The numbers NMR and NMZ (and the corresponding positions of the mesh points) together with the spline orders KR and KZ control the errors in the approximate solution as far as the spatial approximation

is concerned. For heat conduction problems with very smooth data and at points not near the corners, one can expect the spatial error to behave as $O(h^k)$ asymptotically as $h \to 0$. Here h is an appropriate measure of the mesh width and k = KR = KZ. Thus, if one is familiar with the mesh requirements for a second order accurate finite difference scheme in order to achieve a given accuracy, then for a cubic spline (K=4) one would start by trying to use the square root of the number of points needed for the second order method. Of course, this estimate applies only for h rather small, e.g. $h \sim \frac{1}{100}$ one might use $h \sim \frac{1}{10}$ with a cubic. For illustrative purposes we use

NMR=4, NMZ=1,

10. Additional non-interface mesh points

These are the NMR and NMZ mesh points discussed above.

RMESH=0.1,0.25,0.5,1.0,
ZMESH=0.5,

11. Continuity indices at non-interface mesh points

Not needed
(Default)

12. Quadrature order

NQR(NQZ) is the number of quadrature points used in each r(z) mesh interval. The choice NQR=KR and NQZ=KZ is a conservative choice.
NQR=4, NQZ=4,

13. Index for algebraic boundary conditions

ALGBCS=T, (Default) indicates that we will use the boundary conditions as they are given. If we were to use the differential version, we would set ALGBCS=F, then differentiate the essential boundary condition (7.4) with respect to time; thereby generating

$$\frac{\partial \phi}{\partial t}(z,t) = \theta(z),$$

and use the user routine BRHODT to provide $\theta(z)$. With ALGBCBS=T, we use the user routine BRHO to provide $t\theta(z)$.

14. Conservative form index for the convection term

    Since the convection term is zero, use the default value.
    CØNSRV=T, (Default)

15. Material table

    Not needed (Default)

16. Selection of calls to the calculation drivers
    INITSW=T, GUESSW=F, STEDSW=F, TRANSW=T,

    Since the initial condition for this problem is a zero temperature, we could have set the initial coefficients to zero in Namelist DATA rather than having INITSW=T. GUESSW=F, because we have initial conditions and STEDSW=F, because this is not a steady-state problem.

17. Species dependent heat capacity term

    Since the coefficient of $\frac{\partial u}{\partial t}$ in Eq. (7.1.1) is identically one, we set
    IREVLA(1)=F, IRHØ(1)=F, (Defaults)

18. Frequency of output from a steady-state calculation

    Not applicable.

19. Indicator for restarting a STEADY calculation

    Not applicable.

20. Indicator for restarting a TRANSIENT calculation

    Since this is not a restart we use
    ITRARS=F, (Default)

21. Indicator for reading coefficients from unit 10 generated at the conclusion of a previous steady-state calculation

    Not applicable.

22. Number of points in R-direction for user's grid

    IRGRD=4,

23. R coordinates for user's grid

```
RGRID=0.25,0.75,1.25,1.75,
```

24. Number of z coordinates in user's grid

```
JZGRD=2,
```

25. Z coordinates for user's grid

```
ZGRID=0.25,0.75,
```

26. Indicator for an analytic solution

For this problem, we can approximate the solution by means of a double Fourier series.

```
IANAL=T,
```

Using the default options wherever possible, we see that the input for this Namelist GRID requires 10 cards.

Next we consider Namelist DATA. As mentioned before, there are two sets of namelist DATA, the first set is read before a steady-state calculation and the second set is read before a transient calculation. Since we are not doing a steady-state calculation we would fill out the second set of Namelist DATA and ignore the first set. That is, our deck would read:

```
&DATA
&END
&DATA
"To be filled in"
 :
 :
&END
```

From a conceptual point of view this is the proper form for the program; however, the program will also accept (for a transient calculation) all the data in the first namelist, i.e.

```
&DATA
"To be filled in"
 :
 :
&END
&DATA
&END
```

In this sample problem we will use this last form. We now consider the variables to be specified in Namelist DATA.

Namelist DATA

1. Boundary condition switches

Recall that NSI(K) is the switch for species K on side I with

NSI(K) = 0 non-essential b.c. on side I for species K
1 essential b.c. on side I for species K
-1 no b.c. on side I for species K

In this problem we have essential conditions on all four sides.
NS1(1)=1, NS2(1)=1, NS3(1)=1, NS4(1)=1,

2. Boundary value coefficients (MAXSP,4)

ALPHA(1,1)=1.0, BETA(1,1)=0.0, GAMMA(1,1)=1.0,
ALPHA(1,2)=1.0, BETA(1,2)=0.0, GAMMA(1,2)=0.0,
ALPHA(1,3)=1.0, BETA(1,2)=0.0, GAMMA(1,3)=0.0,
ALPHA(1,4)=1.0, BETA(1,4)=0.0, GAMMA(1,4)=0.0,

3. Boundary h functions

Not applicable (Default)

4. Gap coefficients

Not applicable (Default)

5. Reaction coefficients, first order

Not applicable (Default)

6. Reaction coefficients, second order

Not applicable (Default)

7. Initial time for the start of the transient calculation.

For this problem we use 0.0, i.e.
TO=0.0, (Default)

8. Output time control for a transient calculation

NUTOUT is the number of output time values including the initial and final times.
NUTOUT=5,
UTOUT(I), $1 \leq I \leq$ NUTOUT is the array of output times. For this problem we use UTOUT=0.0,0.25,0.5,0.75,3.0,

Each time interval defined by the array UTOUT can be subdivided into NUFREQ subintervals. The output will then be provided at each time point of this fine grid. For this problem we use
NUFREQ=1, (Default)

9. Internal output control switches

Not applicable (Default)

10. Initial spline coefficients

Since B-splines are not interpolatory splines, it is not, in general, a simple matter to select the initial spline coefficients. Thus in general, one would use the least squares fit provided by setting INITSW=T, in Namelist GRID. However, in some situations, it is convenient to provide the initial spline coefficients. For example, in this problem, the initial data is zero everywhere; thus we could have set W=56*0.0, and INITSW=F,.
Note that for this problem there are 56 variables; however, we could have used any number I, $56 \leq I \leq$ MXNVAR. The use of initial spline coefficients is not recommended for those who are not familiar with B-splines and the ordering procedure used in this program, c.f. section 4.10. For this problem, we use the default
W=1.0,... (Default)

11. ODE solver control parameters

EPS is the error control parameter used in the GEAR solver to control the error in the time integration. For this problem we used
EPS=1.D-6,
HINIT is the initial time step used in the GEAR solver. If this initial step is too large relative to the size of EPS, the GEAR solver will reduce the initial step size. For this problem we use
HINIT=1.D-5,

12. Maximum order for the time integration used in GEAR

MXGØRD=5, (Default)
Note that one must have $1 \leq$ MXGØRD ≤ 5.

13. Graphics output switch

If any graphical output is desired at the end of this computer run or at some later time, then when this switch is true the data for the graphics programs will be written on unit 12 for graphical purposes.
GRAPH=T,

For this test problem, we need 10 cards in Namelist DATA if we use the default values wherever possible.

Next we will consider the cards which are needed for the user-supplied subroutines. We shall assume that the user starts from the model subroutines as described in sections 4.1-4.10.

Subroutine RHOCP

Since the coefficient of $\frac{\partial u}{\partial t}$ in Eq. (7.1.1) is identically equal to 1, we need only let

```
RC=1.DO
```

Subroutine DIFUSE

The diffusivity is identically equal to 1; thus we set

```
DIFUR=1.DO
DIFUZ=1.DO
```

Subroutine VEL

There is no convective term in this problem; hence

```
VELR=0.DO
VELZ=0.DO
```

Subroutine EXTSRC

The distributed source is identically zero; hence

```
VV=0.DO
```

Subroutine FDEXTU

Since the distributed source is zero, the Frechet derivatives are identically zero.

```
UU(1)=0.DO
UUR(1)=0.DO
UUZ(1)=0.DO
```

(This problem has only one species.)

Subroutine INDATA

The initial data is identically zero; thus

```
UU=0.DO
```

Subroutine BRHO

From Eq. (7.1.2), we have $u(0,z,t) = t\theta(z)$ on side 1; thus for side 1 we have

```
101 CØNTINUE
    RHOV=2.DO*T*XX
    IF(XX .GT. 0.5DO) RHOV=2.DO*T*(1.DO-XX)
    RETURN
```

From Eq. (7.1.2), $u \equiv 0$ on the remaining sides; thus we use

```
RHOV=0.DO
```

for sides 2, 3, and 4.

Subroutine BRHODT

Since we are using the algebraic boundary conditions, this routine can be a dummy routine.

Subroutine ANAL

For this problem we can provide an alternative approximate solution to this problem by using the Green's function for Eq. (7.1.1) (cf. [5]). The Green's function for the equation

$$\frac{\partial u}{\partial t} = \Delta u$$

with zero boundary values over a two-dimensional rectangle $R = [0,a]\times[0,b]$ is given by

$$G(x,y: x',y',t-\tau) = \frac{4}{ab} \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} e^{-\sigma(t-\tau)} \sin\frac{m\pi x}{a} \sin\frac{m\pi x'}{a} \sin\frac{n\pi y}{b} \sin\frac{n\pi y'}{b}$$

where $\sigma = \pi^2\left(\frac{m^2}{a^2} + \frac{n^2}{b^2}\right)$.

The solution to the problem

$$\frac{\partial u}{\partial t} = \Delta u \text{ in } R, \quad u = \phi \text{ on } \partial R, \quad u = f \text{ at } t = 0,$$

is given by

$$u(x,y,t) = \iint_R [G(x,y: x',y',t]f(x',y')dx'dy' + \int_0^t \int_{\partial R} \phi(x',y',\tau) \frac{\partial G}{\partial n} d\sigma d\tau$$

where n is the inward directed normal to ∂R.

For this problem, $f \equiv 0$, $a = 2$, $b = 1$, and

$$\phi = \begin{cases} t\theta(y) & \text{on side 1,} \\ 0 & \text{on sides 2, 3, and 4,} \end{cases}$$

where

$$\theta(y) = \begin{cases} 2y, & 0 \le y \le \tfrac{1}{2} \\ 2(1-y), & \tfrac{1}{2} \le y \le 1 \end{cases}$$

Hence the solution for this problem has the representation

$$u = \pi \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} m \sin \frac{\pi m x}{2} \sin n\pi y \int_0^t \tau\, e^{-\sigma(t-\tau)} d\tau \int_0^1 \theta(y') \sin n\pi y' dy' .$$

We have

$$\int_0^t \tau\, e^{-\sigma(t-\tau)} d\tau = \frac{1}{\sigma^2} (e^{-\sigma t} + \sigma t - 1) ,$$

$$\int_0^1 \theta(y') \sin n\pi y' dy' = \frac{4}{\pi^2 n^2} \sin \frac{n\pi}{2}, \quad n \ge 1 ;$$

thus

$$(7.6) \qquad u = \frac{4}{\pi} \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \frac{m}{n^2} \frac{1}{\sigma^2} (e^{-\sigma t} + \sigma t - 1) \sin \frac{n\pi}{2} \sin \frac{m\pi x}{2} \sin n\pi y .$$

This double series was implemented in subroutine ANAL. The question of when to truncate the series was settled as follows. For this problem, three decimal place accuracy is enough to determine the error between the series solution and the approximate solution. Let N denote the upper limit for the n index and M the upper limit for the m index. For t=3, x=0.25, y=0.25, the series (7.6) was evaluated for various computations of N and M with the results shown in the following table.

N	10	15	50	15	15	15	15
M	1000	1000	1000	3000	5000	7500	10000
u	.75678	.75675	.75675	.75515	.75483	.75429	.75411

On the basis of these results, we selected the series solution with N=15 and

M=5000 as being almost accurate to three decimal places.

For this problem, both of the arrays AL and GPW in the Master Driver routine were of dimension 4100. The program was run on an IBM 370/195 in 410K bytes of fast memory. The CPU time for this problem was about 117 seconds of which 52 seconds were used in computing the series solution leaving 65 seconds for the computation of the approximate solution on the time interval [0,3].

The following pages contain the printed output from the computational phase of DISPL.

```
STORAGE MAXIMA FOR THIS COMPILATION :
MAXBRK          30
MAXSP            2
MAXTOD           4
MAXK             4
MXNRNZ         100
MXNVAR         100
MAXGAP           2
MAXHOT          40
MXRGBD          20
MXZGBD          20
```

```
 READING NAMELIST  GRID


SPLINE ORDER IN R DIRECTION KR =  4
SPLINE ORDER IN Z DIRECTION KZ =  4
NUMBER OF SPECIES  NSPEC =   1
 LEFT BOUNDARY   RLOW =    0.0
 RIGHT BOUNDARY  RUP  =    0.20000000000000000D+01
 LOWER BOUNDARY  ZLOW =    0.0
 UPPER BOUNDARY  ZUP  =    0.10000000000000000D+01
GEOMETRY INDICATOR  DELTA =  0
NUMBER OF INTERFACES IN R AND Z  NTIR =  0 NTIZ =  0
NUMBER OF NON-INTERFACE MESH POINTS IN R AND Z   NMR =  4   NMZ =   1
 INITIAL CONTINUITY AT MESH POINTS   CONTR =  3          CONTZ =  3
NON-INTERFACE R MESH POINTS
I =  1 RMESH(I) =    0.10000000000000000D+00
I =  2 RMESH(I) =    0.25000000000000000D+00
I =  3 RMESH(I) =    0.50000000000000000D+00
I =  4 RMESH(I) =    0.10000000000000000D+01
NON-INTERFACE Z MESH POINTS
I =  1 ZMESH(I) =    0.50000000000000000D+00
QUADRATURE ORDER FOR R DIRECTION  NQR =      4
QUADRATURE ORDER FOR Z DIRECTION NQZ =      4
CONSERVATIVE FORM INDEX FOR CONVECTION TERM CONSRV =  T
INDEX FOR ALGEBRAIC BOUNDARY CONDITIONS ALGBCS =  T
LOGICAL SWITCHES TO CONTROL PROGRAM
INDICATOR FOR STEADY STATE COMPUTATION STEDSW =  F
INDICATOR FOR TRANSIENT COMPUATION     TRANSW =  T
INDICATOR FOR INITIAL FIT OF DATA      INITSW =  T
INDICATOR FOR STEADY STATE RESTART     ISTDRS =  F
INDICATOR FOR TRANSIENT RESTART        ITRARS =  F
INDICATOR FOR READING STEADY STATE COEFFICIENTS FROM UNIT 10
                                       DUMPSW =  F
INDICATOR FOR ANALYTIC SOLUTION        IANAL  =  F
IBEVLA(K) IS A SWITCH FOR INDICATING THAT THE COEFFICIENT OF THE TIME DERIVATIVE IS NOT IDENTICALLY ONE
IBHO(K)   IS A SWITCH FOR INDICATING THAT THE COEFFICIENT OF THE TIME DERIVATIVE IS IDENTICALLY ZERO
IBEVLA( 1) =  F
IBHO( 1) =  F
NUMBER OF USER SUPPLIED POINTS IN R DIRECTION  IRGRD =  4
RGRID( 1) =    0.25000000000000000D+00
RGRID( 2) =    0.75000000000000000D+00
RGRID( 3) =    0.12500000000000000D+01
RGRID( 4) =    0.17500000000000000D+01
NUMBER OF USER SUPPLIED POINTS IN Z DIRECTION  JZGRD =  2
ZGRID( 1) =    0.25000000000000000D+00
ZGRID( 2) =    0.75000000000000000D+00


 READING NAMELIST  DATA


NVGAP =  0     NHGAP =  0     LR =  5     LZ =  2     NR =  8     NZ =  5

IL(I) =
 4   5   6   7   8

JL(J) =
 4   5

IBEF(I) =
```

```
 1   1   1   1   1

JBEF(J) =  1   1

VERTICAL ORDERING



NI=   5          NJ=   1          NCC=  -6          MBW=  18



NIH=   5          NJH=   1          NCCH=  -5          DM=  18
REQUIRED STORAGE FOR AL   NSTAL =     2200
AVAILABLE STORAGE FOR AL    NAL =     2200
REQUIRED STORAGE FOR PW   NSTPW =     2200
AVAILABLE STORAGE FOR PW    NPW =     2200



SYSTEM SIZE FOR THIS CASE
LB =     5     LZ =     2
NR =     8     NZ =     5
NVAR =   40
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  1
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  2
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  3
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  4
FOR SPECIES NO.  1
 SIDE  1 ALPHA =    0.1000000000000000D+01      BETA =    0.0                    GAMMA =    0.1000000000000000D+01
 SIDE  2 ALPHA =    0.1000000000000000D+01      BETA =    0.0                    GAMMA =    0.1000000000000000D+01
 SIDE  3 ALPHA =    0.1000000000000000D+01      BETA =    0.0                    GAMMA =    0.1000000000000000D+01
 SIDE  4 ALPHA =    0.1000000000000000D+01      BETA =    0.0                    GAMMA =    0.1000000000000000D+01

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  1     NS2=  1     NS3=  1     NS4=  1
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
FOR MATERIAL INDEX  1 AND SPECIES NO.  1 HU1 =    0.1000000000000000D+01      HU3 =    0.1000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
FOR MATERIAL INDEX  1 AND SPECIES NO.  1 HU2 =    0.1000000000000000D+01      HU4 =    0.1000000000000000D+01
ISTDFQ IS PRINTOUT FREQUENCY FOR STEADY STATE COMPUTATION

ISTDFQ =   100
TIME GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  NUTOUT =      5
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL  NUFREQ =  1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES  UTOUT =
   0.0                    0.2500000000000000D+00   0.5000000000000000D+00   0.7500000000000000D+00
   0.3000000000000000D+01
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  T
NUMBER FOR GRAPHS ASSOCIATED WITH THIS RUN
      1
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
LOCAL TEMPORAL ERROR CONTROL  EPS =    0.1000000000000000D-05
```

```
 INITIAL TIME STEP           HINIT =    0.1000000000000000D-04
 MAXIMUM ORDER OF TIME INTEGRATION
                             MXGORD =  5
INITIAL TIME  TO =    0.0
CONTINUITY FOR R AND Z DIRECTIONS MAY HAVE BEEN RESET
CCNTR =  3          CONTZ =  3
DEFAULT INITIAL COEFFICIENTS   W =


   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01

 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  2)-TH RECTANGLE.
(FROM INIFIT)
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  2,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  2,  2)-TH RECTANGLE.
(FROM INIFIT)
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  3,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  3,  2)-TH RECTANGLE.
(FROM INIFIT)
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  4,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
   0.0                      0.0                      0.0                      0.0
```

```
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  4,  2)-TH RECTANGLE.
(FROM INIFIT)
   0.0                   0.0                   0.0                   0.0
   0.0                   0.0                   0.0                   0.0
   0.0                   0.0                   0.0                   0.0
   0.0                   0.0                   0.0                   0.0
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  5,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.0                   0.0                   0.0                   0.0
   0.0                   0.0                   0.0                   0.0
   0.0                   0.0                   0.0                   0.0
   0.0                   0.0                   0.0                   0.0
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  5,  2)-TH RECTANGLE.
(FROM INIFIT)
   0.0                   0.0                   0.0                   0.0
   0.0                   0.0                   0.0                   0.0
   0.0                   0.0                   0.0                   0.0
   0.0                   0.0                   0.0                   0.0


 READING NAMELIST  DATA
```

```
 CHANGES IN NAMELIST DATA MAY HAVE BEEN MADE FOR TRANSIENT
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  1
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  2
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  3
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  4
FOR SPECIES NO.  1
 SIDE  1 ALPHA =    0.1000000000000000D+01      BETA =    0.0                    GAMMA =    0.1000000000000000D+01
 SIDE  2 ALPHA =    0.1000000000000000D+01      BETA =    0.0                    GAMMA =    0.1000000000000000D+01
 SIDE  3 ALPHA =    0.1000000000000000D+01      BETA =    0.0                    GAMMA =    0.1000000000000000D+01
 SIDE  4 ALPHA =    0.1000000000000000D+01      BETA =    0.0                    GAMMA =    0.1000000000000000D+01

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  1     NS2=  1     NS3=  1     NS4=  1
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
FOR MATERIAL INDEX  1 AND SPECIES NO.  1 HU1 =    0.1000000000000000D+01      HU3 =    0.1000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
FOR MATERIAL INDEX  1 AND SPECIES NO.  1 HU2 =    0.1000000000000000D+01      HU4 =    0.1000000000000000D+01
ISTDPQ IS PRINTOUT FREQUENCY FOR STEADY STATE COMPUTATION

ISTDPQ =   100
TIME GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  NUTOUT =     5
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL  NUFREQ =  1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES  UTOUT =
   0.0                    0.2500000000000000D+00   0.5000000000000000D+00   0.7500000000000000D+00
   0.3000000000000000D+01
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  T
NUMBER FOR GRAPHS ASSOCIATED WITH THIS RUN
    1
PRINT SWITCH INDICATORS
IPRSW1 = 0     IPRSW2 = 0     IPRSW3 = 0     IPRSW4 = 0     IPRSW5 = 0
ODE PACKAGE DATA
LOCAL TEMPORAL ERROR CONTROL  EPS =    0.1000000000000000D-05
 INITIAL TIME STEP           HINIT =    0.1000000000000000D-04
 MAXIMUM ORDER OF TIME INTEGRATION
                             MXORD =  5
INITIAL TIME  T0 =    0.0
CONTINUITY FOR R AND Z DIRECTIONS
CONTR =  3          CONTZ =  3
INITIAL COEFFICIENTS FOR TRANSIENT   W =

   0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0
```

```
 BEGIN TRANSIENT SOLUTION


PROUT FOR TIME =     0.0

  W =
   0.0                    0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0                    0.0
   0.0                    0.0                    0.0                    0.0                    0.0

 VALUES OF CONCENTRATIONS U FOR THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TINEX )

APPROXIMATION IS
   0.0                    0.0                    0.0                    0.0

APPROXIMATION IS
   0.0                    0.0                    0.0                    0.0
   TOUT=    0.0


PROUT FOR TIME =     0.2500000000000000D+00

  W =
  -0.72947668440135490-19   0.50000000000000140-01   0.396153846153845 9D+00   0.5000000000000005D-01   0.1356547937945200D-19
   0.1191076786361984D-18   0.5567860934814200D-01   0.3264220664133742D+00   0.5567860934814195D-01  -0.2007154186117187D-19
  -0.3469285311744435D-19   0.5373622275208786D-01   0.2065902784547087D+00   0.5373622275208784D-01  -0.7307272803710924D-20
  -0.87C997472386 C240-20   0.3141263397044821D-01   0.9165137550979925D-01   0.3141263397044821D-01   0.7337447694122858D-20
  -0.1657662927269066D-20   0.6428156872617117D-02   0.1723294690525720D-01   0.6428156872617116D-02  -0.1229377366091023D-20
  -0.1455754191014963D-20  -0.9187225035183389D-03  -0.2253267119225180D-02  -0.9187225035183423D-03   0.1266968551591699D-20
   0.2626484113476297D-20   0.2449097807896525D-03   0.5644143817695680D-03   0.2449097807896548D-03  -0.7778647484426675D-21
  -0.1904932641003882D-20   0.1361843958908679D-20  -0.1070679977371457D-20   0.2756665947182426D-21   0.2674220873769931D-21

 VALUES OF CONCENTRATIONS U FOR THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TINEX )

APPROXIMATION IS
   0.5374667375188498D-01   0.7113500097711235D-02   0.8709248788444497D-03   0.7229221048526679D-04

APPROXIMATION IS
   0.5374667375188499D-01   0.7113500097711235D-02   0.8709248788444502D-03   0.7229221048526637D-04
   TOUT=    0.2500000000000000D+00  HUSED =    0.2380047646046203D-01

ESTIMATED TIME FOR A CALL TO DRIVE    0.0
TIME LEFT     0.1000000000000000D+07


PROUT FOR TIME =     0.5000000000000000D+00

  W =
   0.6651538641758084D-19   0.1000000000000000 9D+00   0.7923076923076917D+00   0.1000000000000002D+00   0.2577480795162574D-19
  -0.1103256441920334D-18   0.1132482529330754D+00   0.6597629282063335D+00   0.1132482529330754D+00  -0.3637526709426631D-19
```

```
   0.3349908575906345D-19   0.1131317100717486D+00   0.4310234465407386D+00   0.1131317100717486D+00  -0.1217076680055513D-19
   0.1370626899562569D-20   0.7130476600993160D-01   0.2071949701790271D+00   0.7130476600993156D-01   0.5764042240925000-20
  -0.8463283978811017D-20   0.1905980936178273D-01   0.5138570865895482D-01   0.1905980936178276D-01  -0.7295569293918845D-20
   0.3070621213883561D-20  -0.1487025291945945D-02  -0.3498566259084060D-02  -0.1487025291945993D-02   0.7816151243049803D-20
  -0.2161831940758551D-20   0.8984924457275406D-03   0.2224464177559582D-02   0.8984924457275646D-03  -0.4864626619952347D-20
   0.7719103621400064D-21  -0.9994207095871014D-21   0.1604448305352406D-20  -0.1454441404481281D-20   0.1681951751928367D-20


 VALUES OF CONCENTRATIONS U FOR THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATION IS
   0.1175448076984768D+00   0.2020884544007021D-01   0.3451868398395700D-02   0.4897781155596646D-03

APPROXIMATION IS
   0.1175448076984769D+00   0.2020884544007020D-01   0.3451868398395706D-02   0.4897781155596608D-03
   TOUT=   0.5000000000000000D+00  HUSED =   0.3911432534183848D-01

ESTIMATED TIME FOR A CALL TO DRIVE    0.0
TIME LEFT    0.1000000000000000D+07



PROUT FOR TIME =    0.7500000000000000D+00

  U =
  -0.1977418591724114D-20   0.1500000000000009D+00   0.1188461538461536D+01   0.1500000000000009D+00  -0.1760813042178002D-19
   0.4353926388106791D-20   0.1708303905405104D+00   0.9931379713289762D+00   0.1708303905405104D+00   0.2935276050054396D-19
  -0.4674495779478280D-20   0.1725712798162247D+00   0.6555772161211933D+00   0.1725712798162249D+00  -0.1075976805031355D-19
   0.6606640786416378D-20   0.1112997102137181D+00   0.3230198394262686D+00   0.1112997102137179D+00   0.1078104296657445D-19
  -0.8768208944807520D-20   0.3188139428508883D-01   0.8605809203200129D-01   0.3188139428508892D-01  -0.2231019557010918D-20
   0.2254229505489684D-20  -0.1846073860617522D-02  -0.4171382537745868D-02  -0.1846073860617628D-02   0.2256543419131450D-20
  -0.1631300911039169D-20   0.1619913629894906D-02   0.4070108170045271D-02   0.1619913629894956D-02  -0.1377060013700826D-20
   0.6668123114862448D-21  -0.6181838477853578D-21   0.7448031945720230D-21  -0.5209269203836307D-21   0.4722984566827527D-21


 VALUES OF CONCENTRATIONS U FOR THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATION IS
   0.1814606219634240D+00   0.3355248075114753D-01   0.6237633840167344D-02   0.9783743121662445D-03

APPROXIMATION IS
   0.1814606219634240D+00   0.3355248075114751D-01   0.6237633840167357D-02   0.9783743121662371D-03
   TOUT=   0.7500000000000000D+00  HUSED =   0.6925024337172310D-01

ESTIMATED TIME FOR A CALL TO DRIVE    0.0
TIME LEFT    0.1000000000000000D+07



PROUT FOR TIME =    0.3000000000000000D+01

  U =
  -0.3926570217124249D-20   0.6000000000000032D+00   0.4753846153846148D+01   0.6000000000000029D+00  -0.6058333156337899D-20
   0.6393364129604478D-20   0.6890738931443021D+00   0.3993525025350625D+01   0.6890738931443014D+00   0.9828333628753606D-20
  -0.1541457364757132D-20   0.7075423176697083D+00   0.2676601933889104D+01   0.7075423176697083D+00  -0.2039267598267599D-20
   0.1921741528659908D-20   0.4712897684736531D+00   0.1365540949599249D+01   0.4712897684736528D+00   0.2024300451500311D-20
  -0.1170071019872095D-20   0.1473433832433697D+00   0.3982948253540327D+00   0.1473433832433695D+00   0.5139466727982166D-21
   0.2161969251250565D-20  -0.4979731249444960D-02  -0.9959221027970155D-02  -0.4979731249444901D-02  -0.5448226486907610D-21
  -0.3856654793720451D-21   0.8154669682890778D-02   0.2079571408537245D-01   0.8154669692890719D-02   0.3361198527513383D-21
   0.1100257572898814D-21  -0.5360808696686850D-22  -0.3674070542647306D-21   0.5922725367915088D-22  -0.1156449240021593D-21
```

```
 VALUES OF CONCENTRATIONS U FOR THE    1 TH SPECIES ON THE USERS GRID
(FROM PRINT VIA TIMEX )

APPROXIMATION IS
   0.7567436817439561D+00    0.1537425918845517D+00    0.3140521739769083D-01    0.5415126751116838D-02

APPROXIMATION IS
   0.7567436817439562D+00    0.1537425918845517D+00    0.3140521739769083D-01    0.5415126751116849D-02
   TOUT=    0.3000000000000000D+01  HUSED =    0.3332200893207061D+00

ESTIMATED TIME FOR A CALL TO DRIVE     0.0
TIME LEFT     0.1000000000000000D+07

****** NORMAL DUMP AT END OF TIMEX
         I =     5          TOUT =    0.3000000000000000D+01


END OF CASE
```

We next consider the use of the graphics packages in connection with this problem. While the cross-section plot program can be used to plot the solution along a straight line in the domain, we will illustrate the use of only CØNTØR and THREED in connection with this problem. (The CSP program will be illustrated in the next example.) In either case the graphics data-set generated on unit 12 during the computational phase must be assigned to this unit for the graphics run. For either case we will need Namelist FORMAT.

Namelist FORMAT

1. Iterative or direct indicator
   Since the ordering of the coefficients differs in the two versions, the graphics program needs to know which ordering to use. In our case the direct version was used.
   ITRTV=0, (Default)

2. Number of curves to be plotted
   We have only one equation and thus
   IGNUM=1, (Default)

3. Number of grid points in each direction for graphical purposes
   This number cannot be larger than the default value NRESD. As the complexity of the surface increases, this number must increase.
   NRESIN=NRESD, (Default) (NRESD=21)

For the CØNTØR program we also need Namelist CNTRIN.

Namelist CNTRIN

1. Vector of species numbers
   This vector has, as its Ith component, the species number of the Ith frame to be contoured. The pattern of plots indicated by the vector is repeated for each time value. In the current case we have only one species and so,
   ISPEC=1, (Default)

2. Real vector of minimum R coordinate for the Ith frame
   We use the same value as RLOW.
   RMIN=0.0, (Default)

3. Real vector of maximum R coordinate for the Ith frame
   We use the same value as RUP.
   RMAX=2.0,

4. Real vector of minimum Z coordinate for the Ith frame
   We use the same value as ZLOW.
   ZMIN=0.0, (Default)

5. Real vector of maximum Z coordinate for the Ith frame
   We use the same value as ZUP.
   ZMAX=1.0, (Default)

6. Logical indicator for doing time plots
   ITIME=F,

7. Number of time values in time plots
   Since we are not doing time plots, it is unnecessary to set this variable.
   NTIME=NRESD (Default) (NRESD=21)

For the THREED program we need:

Namelist DIM3IN

All of the variables described in Namelist CNTRIN are used in this namelist. Also, the following variables are needed.

1. R coordinate of the viewpoint
   The viewpoint is given in absolute coordinates. That is, it is relative to the actual coordinate values used in the graph. It is usually a good idea to first view the surface from a considerable distance. Thus, the viewpoint used was (100.,100.,100.).
   RVIEW=100.0,

2. Z coordinate of viewpoint.
   ZVIEW=100.0,

3. F coordinate of viewpoint
   This is the vertical coordinate of the viewpoint (the solution axis).
   FVIEW=100.0,

4. Lower bound on function axis
   The lower and upper bounds on the function axis can be used to

provide a fixed vertical axis range for several time values (for use in movie generation). For most purposes it is sufficient to use the default value

FMATMN=0.0, (Default)

5. Upper bound on function axis.

FMATMX=1.0, (Default)

The following pages contain selected printout from the execution of CØNTØR, the corresponding plots, selected printout from the execution of THREED, and the corresponding plots. Specifically, for CØNTØR we present output concerning frame 4 (time = .75) and frame 5 (time = 3.0). For THREED we present the corresponding graphs and the printout associated with frame 4 only. The other printout and graphs are not included in the interest of saving space.

```
BEGINNING CONTOUR GRAPHICS PACKAGE

VERSION      NUMBER OF GRAPHS FOR EACH TIME

 ITRTV =      0      IGNUM =      1      NRESIN =     21

USER SUPPLIED FORMAT INSTRUCTIONS

SPECIES         RMIN                  RMAX                  ZMIN                  ZMAX
ITIME =  F
      1
        0.0                  0.20000000E+01       0.0                  0.10000000E+01

NRESR =     21      NRESZ =     21

TIME FOR THIS RUN IS  0.0

          R                   Z                   F


TIME FOR THIS RUN IS  0.25000000E+00

          R                   Z                   F


TIME FOR THIS RUN IS  0.50000000E+00

          R                   Z                   F


TIME FOR THIS RUN IS  0.75000000E+00

          R                   Z                   F


TIME FOR THIS RUN IS  0.30000000E+01

          R                   Z                   F
```

DIVISION

```
X Y SCALE =       0.250   FT. PER INCH

CONTOUR INTERVAL =       0.100

 1 POINT(S)/CARD    1 POINT(S) IN LAST CARD
GRID BACKGROUND= 1
CONTOUR LINE ID= 1
IOP SWITCHES=  1  0  0  0  0  0  0  0  0 13  0  0  0  0  0  0  0  0  0  0

XORIG=  .0            YORIG=  .0              INPUT SCALE IN UNITS/INCH. X=  .2500000      AND Y=  .2500000
 RATIO OF OUTPUT MAP SIZE TO INPUT MAP SIZE. X DIRECTION=  1.000000      AND Y DIRECTION=  1.000000
ZFLAG =  .0
VARIABLE FORMAT=     (1X,3E14.6,A3)

ORIGIN CALCULATED BY CONTOUR PROGRAM
XMIN =   .0                   YMIN =   .0                   XMAX =   8.0000               YMAX =   4.0000
NUMBER OF BASIC DATA POINTS =   441


NEXT PLOTTER TAPE RECORD IS A GRID.
NDEL= 1
ROWS   5
COLS=  9
X SCALE(INCHES) =  1.0000
Y SCALE(INCHES) =  1.0000

NUMBER OF POINTS TO BE PLOTTED =    441

ZMIN,ZMAX= -.31292E-06      .69569
INTERNAL MATRIX ROWS=   21 COLS=   21
NUMBER OF OVERLAYS=     1

CONTOUR STARTS

VECTOR GENERATION COMPLETED, NUMBER OF HEIGHTS=   9
```

```
X Y SCALE =       0.250    FT. PER INCH

CONTOUR INTERVAL =       0.100

 1 POINT(S)/CARD    1 POINT(S) IN LAST CARD
GRID BACKGROUND= 1
CONTOUR LINE ID= 1
IOP SWITCHES=  1  0  0  0  0  0  0  0  0 13  0  0  0  0  0  0  0  0  0  0

XORIG=  .0            YORIG=  .0              INPUT SCALE IN UNITS/INCH. X=  .2500000      AND Y=  .2500000
 RATIO OF OUTPUT MAP SIZE TO INPUT MAP SIZE. X DIRECTION=  1.000000      AND Y DIRECTION=  1.000000
ZFLAG =  .0
VARIABLE FORMAT=     (1X,3E14.6,A3)

ORIGIN CALCULATED BY CONTOUR PROGRAM
XMIN =    .0                  YMIN =    .0                  XMAX =    8.0000              YMAX =    4.0000
NUMBER OF BASIC DATA POINTS =    441


NEXT PLOTTER TAPE RECORD IS A GRID.
NDEL= 1
ROWS   5
COLS=  9
X SCALE(INCHES)=  1.0000
Y SCALE(INCHES)=  1.0000

NUMBER OF POINTS TO BE PLOTTED =     441

ZMIN,ZMAX= -.15423E-05      2.7828
INTERNAL MATRIX ROWS=    21 COLS=    21
NUMBER OF OVERLAYS=      1

CONTOUR STARTS

VECTOR GENERATION COMPLETED, NUMBER OF HEIGHTS=   30
EXIT
```

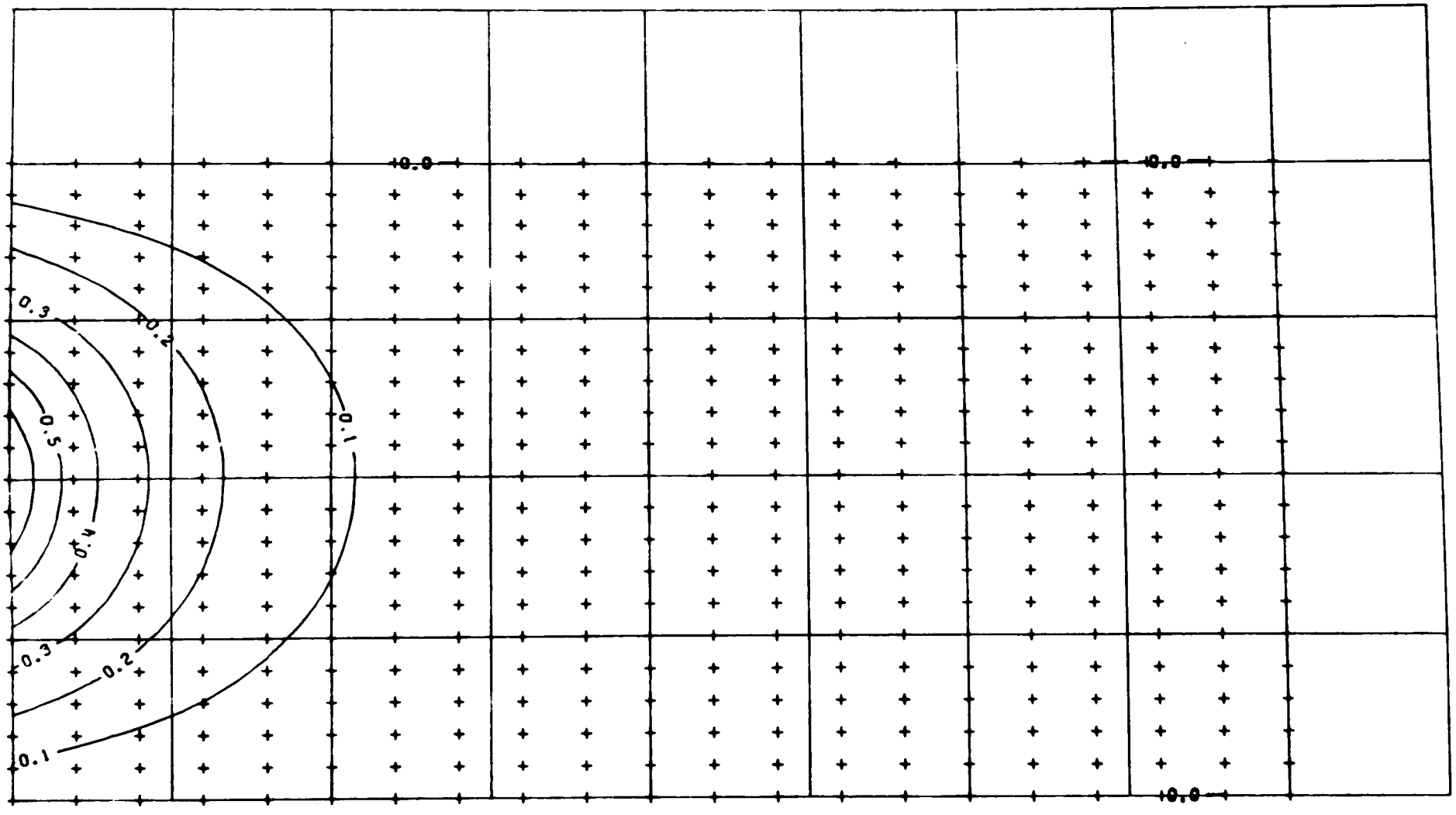


Figure 7.1.1. Contour plot at t = 0.75 (Frame 4)

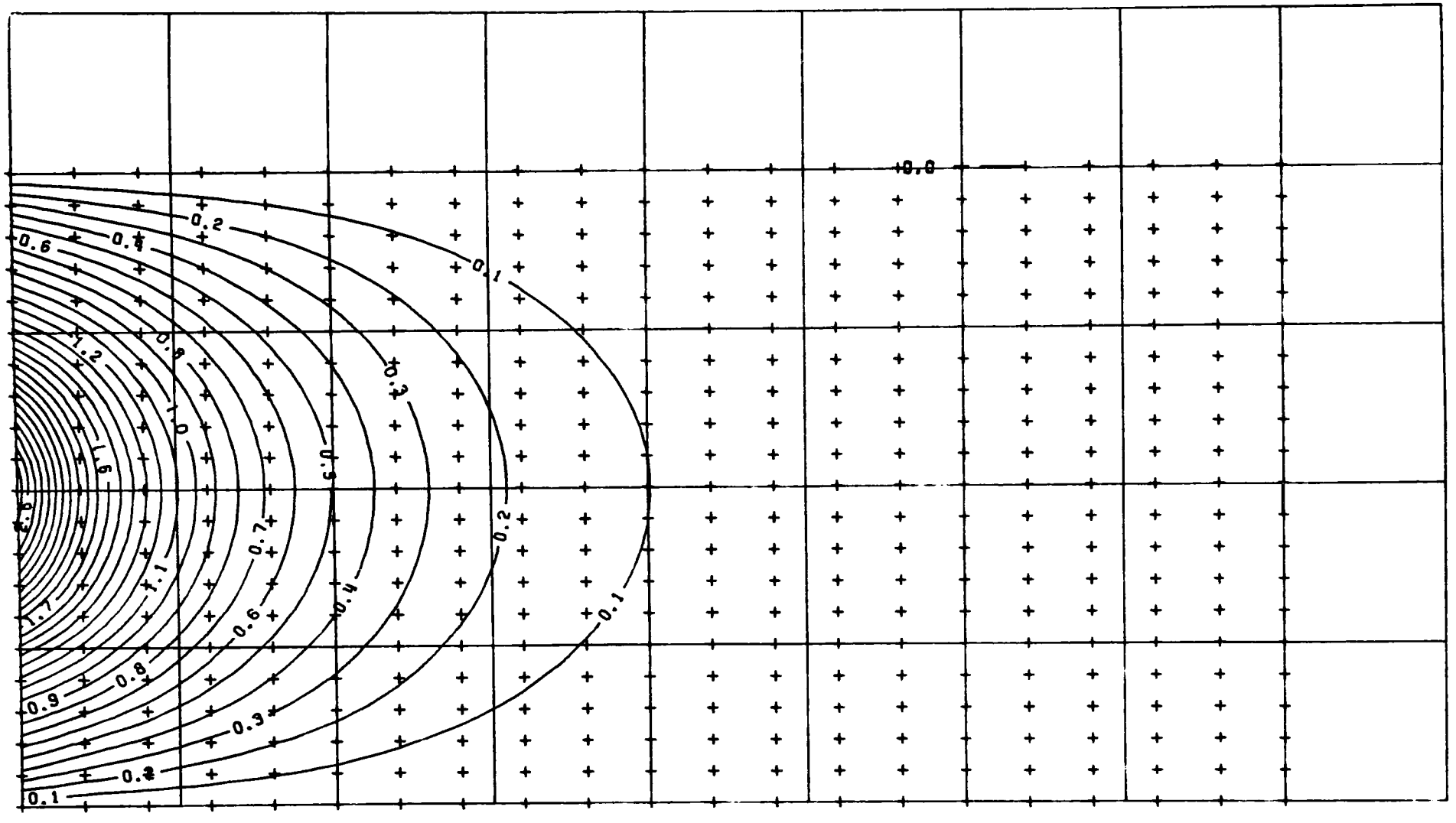


Figure 7.1.2. Contour plot at t = 3.0 (Frame 5)

```
BEGINNING THREE DIMENSIONAL GRAPHICS PACKAGE

VERSION      NUMBER OF GRAPHS FOR EACH TIME

     0                        1


NRESIN =     21

IITME =  F

USER SUPPLIED FORMAT INSTRUCTIONS

SPECIES         RMIN                  RMAX                  ZMIN                  ZMAX
      1
        0.0                  0.20000000E+01        0.0                  0.10000000E+01

NRESR =     21      NRESZ =     21

VIEWPOINT IN ABSOLUTE COORDINATES

         X                     Y                     Z

 0.10000000E+03        0.10000000E+03        0.10000000E+03

USER SUPPLIED F-AXIS RANGE
FMATMN =        0.0                   FMATMX =        0.30000000E+01
```

```
TIME FOR THIS RUN IS  0.75000000E+00
```

```
PLOT NO.   4 WITH THE TITLE
     2-D KINETICS
HAS BEEN COMPLETED.

PLOT ID. HEADS
PLOT 0      11.32.18   FRI 15 APR, 1977    JOB=HEATJD .  ISSCO,   DISSPLA VER 4.11

                    PLOTTING COMMENCING
                    ...................

NO. OF FIRST PLOT  4
```

```
WORKBOX-DIMENSIONS
------------------

   X3DAXIS=    1.00
   Y3DAXIS=    1.00
   Z3DAXIS=    1.00
        IN ABS. 3-D UNITS

VIEWPOINT
---------

   XVU= 1.000E+02
   YVU= 1.000E+02
   ZVU= 1.000E+02
        IN ABS. 3-D UNITS

GRAPH SET-UP ( GRAF3D )
-----------

  ORIGIN
  ------
   X3DORIGIN= 0.0
   Y3DORIGIN= 0.0
   Z3DORIGIN= 0.0

  STEP SIZE
  ---------
   X3DSTP= 5.000E-01
   Y3DSTP= 2.000E-01
   Z3DSTP= 5.000E-01

  MAXIMUM
  -------
   X3DMAX= 2.000E+00
   Y3DMAX= 1.000E+00
   Z3DMAX= 3.000E+00
```

```
LOCATION OF CURRENT PHYSICAL ORIGIN
X=    0.50       Y=    1.12     INCHES
   FROM LOWER LEFT CORNER OF PAGE
```

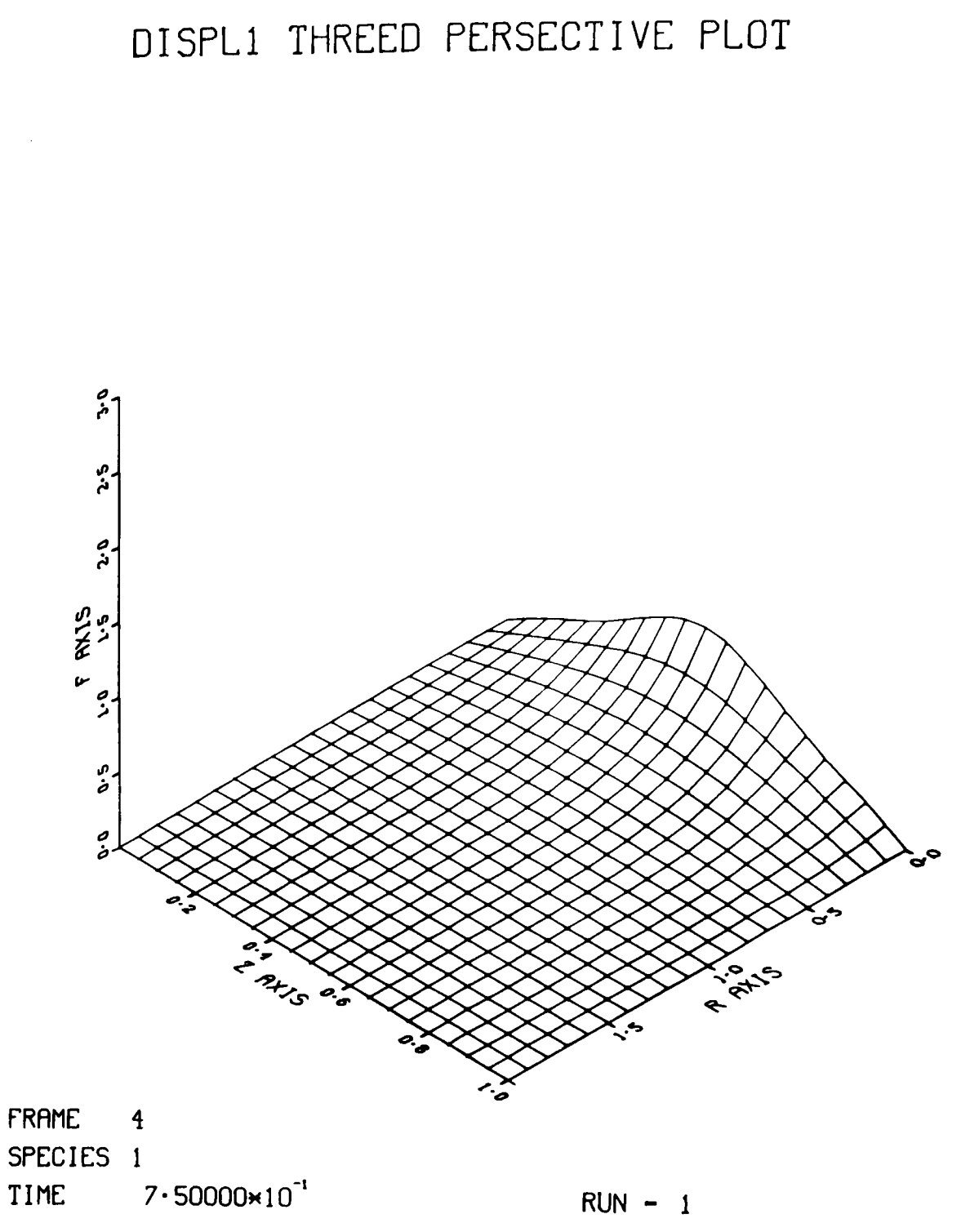


Figure 7.1.3
Three-dimensional perspective surface for heat conduction problem

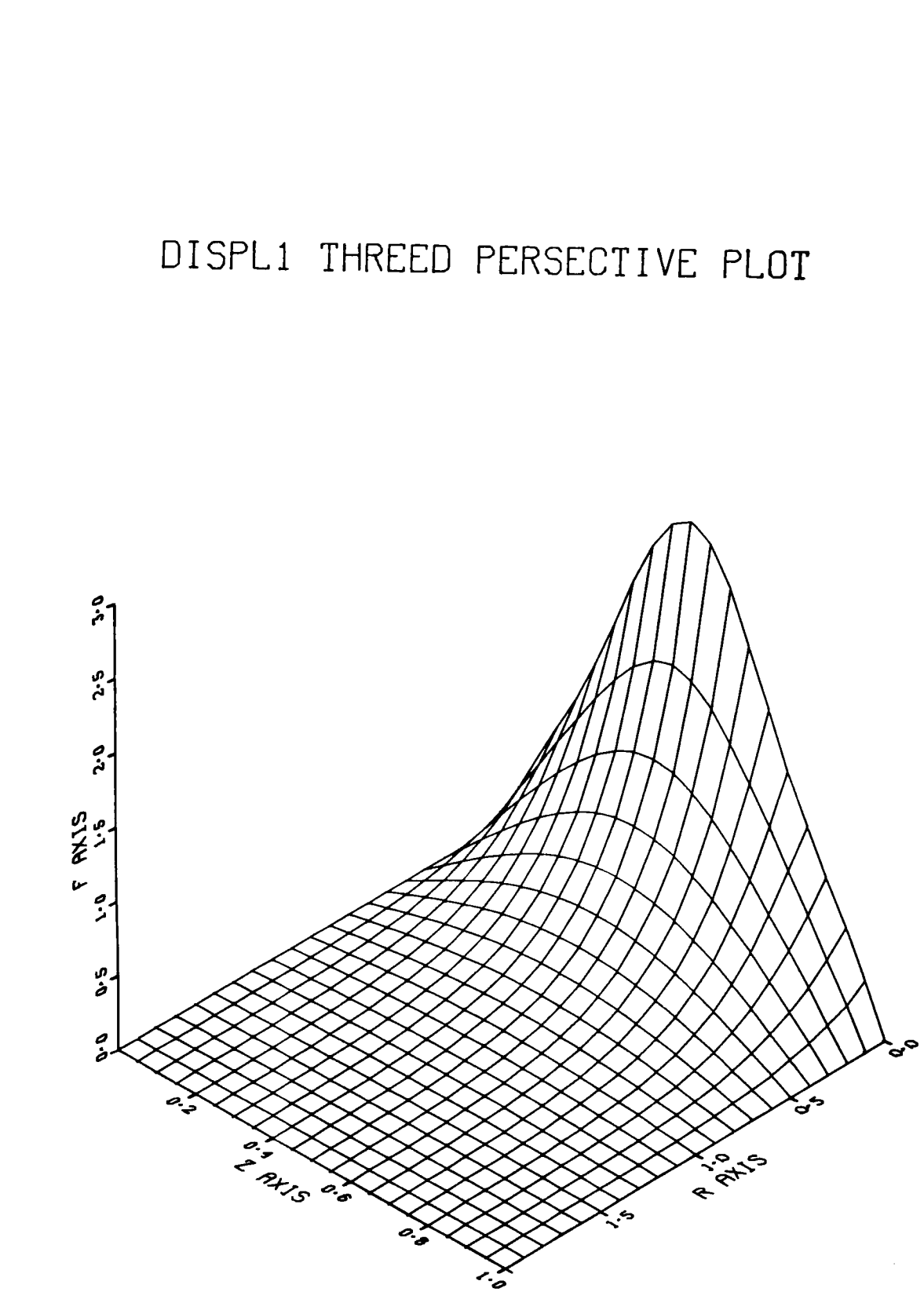


Figure 7.1.4

Three-dimensional perspective surface for heat conduction problem

7.2 Water Hammer

This example involves a hyperbolic system of flow equations for a water hammer problem [6]. The model considers the flow of water from a reservoir down a conduit with a valve at the exit end. The valve is closed at a linear rate and, after closure, the flow reverses.

The continuity and momentum equations form a pair of quasi-linear hyperbolic partial differential equations in two dependent variables, velocity u and pressure head H, and two independent variables, distance along the pipe x and time t. The equations are given by

$$\begin{aligned} &\frac{\partial H}{\partial t} + u\,\frac{\partial H}{\partial x} + \frac{a^2}{g}\,\frac{\partial u}{\partial x} = 0\ , \\ &\frac{\partial u}{\partial t} + u\,\frac{\partial u}{\partial x} + g\,\frac{\partial H}{\partial x} = -\,\frac{u|u|f}{2D}\ , \end{aligned} \tag{7.2.1}$$

where

g = 32.2 ... the acceleration due to gravity.
a = 3963 ... the speed of sound in water.
L = 4253.5.. the length of the horizontal conduit.
D = 3 ..... the diameter of the conduit.
f = 0.019... the friction coefficient.
x ..... the distance along the conduit from the reservoir.
H = H(x,t).. the pressure head in the conduit at x and t.
u = u(x,t).. the velocity in the conduit at x and t.

The valve is located at x = L, and is closed at a linear rate until it is fully closed at $t = t_c = 5.9$; thereafter, the valve remains closed.

The boundary conditions are as follows:

$$\begin{aligned} &H(0,t) = 305.12, \\ &u(L,t) = \tau u_0 \sqrt{\frac{H(L,t)}{H_0}}\ , \end{aligned} \tag{7.2.2}$$

where

$u_0 = 3.5$ ... the steady-state velocity.

$H_0 = 300$ ... the steady state head loss across the valve.

$$\tau = \begin{cases} (1 - t/5.9), & 0 \leq t \leq 5.9 \\ 0, & t > 5.9 \end{cases}$$

The initial conditions are the steady-state conditions, viz.

$$u(x,0) = u_0 = 3.5$$

$$H(x,0) = 305.12 - \frac{(u_0)^2 f\, x}{2gD} \tag{7.2.3}$$

We shall use this problem to illustrate: the procedures for setting up one-dimensional problems, nonlinear boundary conditions, and graphical output for one-dimensional problems.

It is assumed that the reader has already considered sample problem 7.1; therefore, we shall not discuss those items in the namelists which can be ignored for this problem. Recall that this problem is one dimensional and we shall arbitrarily take this dimension as the r direction.

1. Spline Order

   KR=4, KZ=1,

   KZ=1 is one of the parameters which is used in defining a one-dimensional problem. In this problem we will use smooth cubics, so that KR=4, is a default value.

2. Continuity at the mesh points

   The default values are used, i.e.

   CONTR=3, CONTZ=0,

   Note that since KZ=1, CONTZ must be zero.

3. Number of species

   NSPEC=2,

4. Domain

   RUP=4253.5,

   The other values are default values.

9. Total number of non-interface mesh points in each direction

   NMR=9,

   The value NMZ=0, is default and this value should be used for one-dimensional problems in r.

10. Additional non-interface mesh points

   RMESH=425.35, 850.7, 1276.05, 1701.4, 2126.75, 2552.1, 2977.45, 3402.8, 3828.15,

12. Quadrature Order

   NQR=4, NQZ=1,

   NQZ=1 should be used for a one-dimensional problem in the r direction.

13. Index for algebraic boundary conditions

   ALGBCS=T,

   This is the default value; however, it is worth noting that for boundary conditions which are not of the form $u_m = \rho_m$ where ρ_m is a known function of t and x only, the concept of a time derivative of the boundary conditions is not feasible in this program. Hence for non-standard and/or nonlinear boundary conditions, one must use ALGBCS=T,

22. Number of points in the r-direction for the user's grid

   IRGRD=6,

23. R coordinates for the user's grid

   RGRID=0.0, 850.7, 1701.4, 2552.1, 3402.8, 4253.5,

Next we consider Namelist DATA.

1. Boundary condition switches

   NS1(1)=1, NS3(1)=-1,

   NS1(2)=-1, NS3(2)=1,

   Here we have selected the head pressure as the first species and the velocity as the second species. The default values NS2(i) = NS4(i) = 0 for i = 1,2 are needed for a one-dimensional problem in r.

2. Boundary value coefficients

ALPHA(1,1)=1.0, BETA(1,1)=0.0, GAMMA(1,1)=1.0,
ALPHA(1,3)=1.0, BETA(1,3)=0.0, GAMMA(1,3)=1.0,
ALPHA(2,1)=1.0, BETA(2,1)=0.0, GAMMA(2,3)=1.0,
ALPHA(2,3)=1.0, BETA(2,3)=0.0, GAMMA(2,3)=1.0,
The default values on sides 2 and 4 must be used for one-dimensional problems in r.

8. Output time control

NUTOUT=21,
UTOUT=0.0, 0.429, 0.859, 1.288, 1.717, 2.147, 2.576, 3.005, 3.435, 3.864, 4.293, 4.723, 5.152, 5.581, 5.841, 5.9, 6.01, 6.44, 6.869, 7.298, 7.728,
If one were to use, for example, NUTOUT=8 in conjunction with this UTOUT array, the program would integrate the equations from t = 0.0 to t = 3.005 with output at each of the first eight times in the UTOUT array. If one wished to restart from t = 3.005 to go to t = 7.728, for example, then one would use NUTOUT=21 with this same UTOUT array. Thus NUTOUT is the number of output times desired counting from the first element of the UTOUT array. This is true even on a restart.

11. ODE Solver control parameters

For this problem we used
EPS = 1.D-6, HINIT = 1.D-6,

This completes the input for the two namelists for this water hammer problem. Next we consider the user-supplied subroutines. Again we assume that the user starts from the model subroutines described in section 4.

Subroutine RHOCP

For both species, the coefficient of the time derivative term is identically one; thus we use

RC=1.D0

Subroutine DIFUSE

There are no diffusion terms in this problem; hence

DIFUR=0.D0
DIFUZ=0.D0

Subroutine VEL

This problem was run with the convection term explicitly displayed and also with the convection term grouped with the distributed term. In the run shown here we have used the convection term grouped with the distributed source term. Thus we have:

```
VELR=0.D0
VELZ=0.D0
```

Note that the use of VELZ=0.D0 and DIFUZ=0.D0 are part of the defining relations for a problem which is one dimensional in the r direction.

Subroutine EXTSRC

The first species is the pressure and the second species is the velocity. Since we have grouped the convective terms with the distributed source terms, we have from Eq. (7.2.1),

1st species:

$$VV = -u \frac{\partial H}{\partial x} - \frac{a^2}{g} \frac{\partial u}{\partial x} = \text{-SPDEN(2)*SPDENR(1)-ASDG*SPDENR(2)}$$

2nd species:

$$VV = -u \frac{\partial u}{\partial x} - g \frac{\partial H}{\partial x} - \frac{u|u|f}{2D} = \text{-SPDEN(2)*SPDENR(2)-G*SPDENR(1)}$$

```
-F*SPDEN(2)*DABS(SPDEN(2))/2.D0*D
```

where

```
G = 32.2D0
D = 3.D0
F = 0.019D0
ASDG = (3963.D0**2)/G
```

Subroutine FDEXTU

Recall that, for each species, this routine calculates the Frechet derivative of the distributed source with respect to u(k'), $\frac{\partial u}{\partial r}(k')$, and $\frac{\partial u}{\partial z}(k')$ for k' = 1,NSPEC.

Consider the 1st species. The source term is

$$VV = -u \frac{\partial H}{\partial x} - \frac{a^2}{g} \frac{\partial u}{\partial x} = \text{-SPDEN(2)*SPDENR(1)-ASDG*SPDENR(2).}$$

Thus

$$\frac{\partial VV}{\partial u(1)} = 0,$$

i.e.

```
UU(1) = 0.D0.
```

$$\frac{\partial VV}{\partial u(2)} = -\frac{\partial H}{\partial x},$$

i.e.

UU(2) = -SPDENR(1).

$$\frac{\partial VV}{\partial u_r(1)} = -u, \quad (u_r(1) = \frac{\partial H}{\partial x}, \; u_r(2) = \frac{\partial u}{\partial x})$$

i.e.

UUR(1) = -SPDEN(2).

$$\frac{\partial VV}{\partial u_r(2)} = -a^2/g,$$

i.e.

UUR(2) = -ASDG.

This is a one-dimensional problem in r; hence

UUZ(1)=0.D0,
UUZ(2)=0.D0.

Next, consider the 2nd species.

$$VV = -u\frac{\partial u}{\partial x} - g\frac{\partial H}{\partial x} - \frac{u|u|f}{2D}.$$

Thus

$$\frac{\partial VV}{\partial u(1)} = 0,$$

i.e.

UU(1) = 0.D0.

$$\frac{\partial VV}{\partial u(2)} = -\frac{\partial u}{\partial x} - \frac{|u|f}{D},$$

i.e.

UU(2) = -SPDENR(2)-F*DABS(SPDEN(2))/D.

$$\frac{\partial VV}{u_r(1)} = -g$$

i.e.

UUR(1) = -G.

$$\frac{\partial VV}{\partial u_r(2)} = -u,$$

i.e.

```
UUR(2) = -SPDEN(2)
UUZ(1)=0.DO
UUZ(2)=0.DO
```

Subroutine INDATA

The initial conditions are given by Eq. (7.2.3); thus we have:

```
      G=32.2DO
      ASDG=(3963.DO**2)/G
      D=3.DO
      F=0.019DO
      IF(K .EQ. 2) GØ TØ 10
      UU=305.12DO-F*(3.5DO**2)*RR/(2.DO*D*G)
      RETURN
   10 UU=3.5DO
      RETURN
```

Subroutine BRHO

The boundary conditions are given by Eq. (7.2.2). Using the model subroutine for BRHO described in section 4, we have:

after 101

```
RHOV=305.12DO
```

after 203

```
RHOV=0.DO
TAU=1.DO-(T/TC)
IF(T .LT. TC) RHOV=UO*TAU*DSQRT(SPDEN(1)/HO)
RETURN
```

where

```
TC=5.9DO
HO=300.DO
UO=3.5DO
```

In the Master Driver, both AL and GPW were of dimension 784 (AL could have been half this size). The water hammer problem was run on an IBM 370/195. With the same macros as used in the first problem, this problem used 356K bytes of fast memory and ran with 42 seconds for the CPU time. In [6], this problem was solved by the method of characteristics and the results are shown in Table 7.2.1. Since we have selected the output grid and the output times to be the same as those in Table 7.2.1, the results can be compared and the agreement is seen to be excellent.

The following pages contain the printed output from the computational phase of DISPL.

Table 7.2.1

Output from method of characteristic solution

TIME	TAU	X/L=	0.	.2	.4	.6	.8	1.
.000	1.000	H=	305.12	304.10	303.07	302.05	301.02	300.00
		V=	3.50	3.50	3.50	3.50	3.50	3.50
.429	.927	H=	305.12	304.10	303.07	302.05	310.30	318.93
		V=	3.50	3.50	3.50	3.50	3.42	3.35
.859	.854	H=	305.12	304.10	312.30	320.90	329.92	339.37
		V=	3.50	3.50	3.43	3.35	3.27	3.18
1.288	.782	H=	305.12	322.66	331.84	341.24	351.11	361.43
		V=	3.35	3.35	3.27	3.18	3.09	3.00
1.717	.709	H=	305.12	324.44	343.79	363.21	373.98	385.24
		V=	3.04	3.03	3.02	3.01	2.91	2.81
2.147	.636	H=	305.12	326.15	347.20	368.33	389.56	410.93
		V=	2.70	2.69	2.68	2.66	2.64	2.61
2.576	.563	H=	305.12	327.98	350.87	373.83	392.68	410.99
		V=	2.33	2.33	2.32	2.30	2.30	2.31
3.005	.491	H=	305.12	329.94	350.59	370.63	390.06	408.83
		V=	1.94	1.93	1.95	1.97	1.99	2.00
3.435	.418	H=	305.12	322.97	344.33	364.97	384.88	404.01
		V=	1.58	1.58	1.61	1.64	1.67	1.70
3.864	.345	H=	305.12	322.32	339.43	356.38	376.65	396.00
		V=	1.29	1.29	1.30	1.31	1.35	1.39
4.293	.272	H=	305.12	321.34	337.46	353.37	368.98	384.19
		V=	1.02	1.02	1.03	1.04	1.06	1.08
4.723	.200	H=	305.12	319.96	334.68	349.16	366.35	383.85
		V=	.76	.77	.78	.79	.79	.79
5.152	.127	H=	305.12	318.11	334.02	350.45	367.36	384.77
		V=	.54	.54	.53	.52	.51	.50
5.581	.054	H=	305.12	322.63	337.80	353.71	370.34	387.70
		V=	.29	.29	.27	.25	.23	.22
6.010	.000	H=	305.12	323.23	341.44	359.88	376.26	384.48
		V=	.00	.00	-.00	-.01	-.04	.00
6.440	.000	H=	305.12	324.27	343.55	353.99	356.04	358.51
		V=	-.30	-.30	-.31	-.25	-.12	.00
6.869	.000	H=	305.12	310.74	319.73	322.91	323.37	323.51
		V=	-.62	-.55	-.42	-.29	-.14	.00
7.298	.000	H=	305.12	292.17	284.94	286.29	287.08	287.33
		V=	-.53	-.54	-.46	-.30	-.15	.00
7.728	.000	H=	305.12	287.65	272.49	256.61	248.66	249.08
		V=	-.29	-.29	-.27	-.25	-.16	.00

```
STORAGE MAXIMA FOR THIS COMPILATION :
MAXBRK          30
MAXSP            2
MAXTOD           4
MAXK             4
MXNRNZ         100
MXNVAR         100
MAXNOT          40
MXRGRD          20
MXZGRD          20
```

```
 READING NAMELIST  GRID

KP =  4    KZ =  1
NTIR =  0 NTIZ =  0
NSPEC =  2
RLOW =    0.0                                  RUP =    0.42535000000000000+04
ZLOW =    0.0                                  ZUP =    0.10000000000000000+01
NMR =  9     NMZ =  0
 INITIAL CONTP = 13          INITIAL CONTZ = 13
GEOMETRY INDICATOR =  0
ADDITIONAL R GRID POINTS
I =  1 RMESH(I) =    0.42535000000000000+03
I =  2 RMESH(I) =    0.85070000000000000+03
I =  3 RMESH(I) =    0.12760500000000000+04
I =  4 RMESH(I) =    0.17014000000000000+04
I =  5 RMESH(I) =    0.21267500000000000+04
I =  6 RMESH(I) =    0.25521000000000000+04
I =  7 RMESH(I) =    0.29774500000000000+04
I =  8 RMESH(I) =    0.34028000000000000+04
I =  9 RMESH(I) =    0.38281500000000000+04
NO ADDITIONAL Z GRID POINTS
QUADRATURE ORDER FOR R DIRECTION     4
QUADRATURE ORDER FOR Z DIRECTION     1
MATERIAL TABLE IS GIVEN AS MATL(RINDEX,ZINDEX)
MATERIAL TABLE FOR ZINDEX =  1
 1
CONSRV=  F  0
ALGBCS=  T
LOGICAL SWITCHES TO CONTROL PROGRAM
STFDSW =  F      GUESSW =  F      TRANSW =  T      INITSW =  T
ISTDFS =  F          ITRARS =  F          IANAL =  F
DUMPSW =  F
IREVLA( 1) =  F
IREVLA( 2) =  F
ORDER OF SPLINE DERIVATIVES COMPUTED IS  0
NUMBER OF USER SUPPLIED POINTS IN R DIRECTION  6
RGRID( 1) =       0.0
RGRID( 2) =       0.85070000D+03
RGRID( 3) =       0.17014000D+04
RGRID( 4) =       0.25521000D+04
RGRID( 5) =       0.34028000D+04
RGRID( 6) =       0.42535000D+04
NUMBER OF USER SUPPLIED POINTS IN Z DIRECTION  1
ZGRID( 1) =       0.50000000D+00


 READING NAMELIST  DATA


CONTR TOO HIGH, BEING RESET TO      4-1

CONTZ TOO HIGH, BEING RESET TO      1-1

NVGAP =  0     NHGAP =  0     LR = 10     LZ =  1     NR = 13     NZ =  1

IL(I) =
 4   5   6   7   8   9  10  11  12  13

JL(J) =
```

```
 1

IREF(I) =
 1   1   1   1   1   1   1   1   1   1

JREF(J) =  1
MLTAB( 1, 1) =  1
MLTAB( 2, 1) =  1
MLTAB( 3, 1) =  1
MLTAB( 4, 1) =  1
MLTAB( 5, 1) =  1
MLTAB( 6, 1) =  1
MLTAB( 7, 1) =  1
MLTAB( 8, 1) =  1
MLTAB( 9, 1) =  1
MLTAB(10, 1) =  1

HORIZONTAL ORDERING


NI=   2          NJ=  26          NCC= -28          MBW=   7


NIH=   1          NJH=  13          NCCH= -13          DM=   3


SYSTEM SIZE FOR THIS CASE
LR =    10      LZ =     1
NF =    13      NZ =     1
NVAR =   26

 THIS IS THE DIRECT VERSION

 THIS VERSION DOES NOT REQUIRE BOUNDARY CONDITIONS ON EVERY SIDE
FOR SPECIES NO.  1
 SIDE  1 ALPHA =   0.10000000D+01      BETA =   0.0                GAMMA =   0.10000000D+01
 SIDE  2 ALPHA =   0.0                 BETA =  -0.10000000D+01     GAMMA =   0.0
 SIDE  3 ALPHA =   0.10000000D+01      BETA =   0.0                GAMMA =   0.10000000D+01
 SIDE  4 ALPHA =   0.0                 BETA =   0.10000000D+01     GAMMA =   0.0
FOR SPECIES NO.  2
 SIDE  1 ALPHA =   0.10000000D+01      BETA =   0.0                GAMMA =   0.10000000D+01
 SIDE  2 ALPHA =   0.0                 BETA =  -0.10000000D+01     GAMMA =   0.0
 SIDE  3 ALPHA =   0.10000000D+01      BETA =   0.0                GAMMA =   0.10000000D+01
 SIDE  4 ALPHA =   0.0                 BETA =   0.10000000D+01     GAMMA =   0.0

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  1     NS2=  0     NS3= -1     NS4=  0
 FOR SPECIES NO.  2     NS1= -1     NS2=  0     NS3=  1     NS4=  0
CONVECTION VELOCITY IN P DIRECTION    0.0
CONVECTION VELOCITY IN Z DIRECTION    0.0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
SPECIES NO.  1 MATERIAL INDEX  1 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
SPECIES NO.  2 MATERIAL INDEX  1 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
SPECIES NO.  1 MATERIAL INDEX  1 HU2 =    0.10000000000000000D+01      HU4 =    0.10000000000000000D+01
SPECIES NO.  2 MATERIAL INDEX  1 HU2 =    0.10000000000000000D+01      HU4 =    0.10000000000000000D+01
```

```
REACTION RATES
FIRST ORDER RATES
CK INTO  1 FROM  1 IS     0.0
CK INTO  2 FROM  1 IS     0.0
CK INTO  1 FROM  2 IS     0.0
CK INTO  2 FROM  2 IS     0.0
SECOND ORDER REACTION RATES ARE
CKK INTO K =  1 FOR KP =  1 INTO KPP =  1          CKK( 1, 1, 1) =     0.0
CKK INTO K =  1 FOR KP =  2 INTO KPP =  1          CKK( 1, 2, 1) =     0.0
CKK INTO K =  1 FOR KP =  1 INTO KPP =  2          CKK( 2, 1, 1) =     0.0
CKK INTO K =  1 FOR KP =  2 INTO KPP =  2          CKK( 2, 2, 1) =     0.0
CKK INTO K =  2 FOR KP =  1 INTO KPP =  1          CKK( 1, 1, 2) =     0.0
CKK INTO K =  2 FOR KP =  2 INTO KPP =  1          CKK( 1, 2, 2) =     0.0
CKK INTO K =  2 FOR KP =  1 INTO KPP =  2          CKK( 2, 1, 2) =     0.0
CKK INTO K =  2 FOR KP =  2 INTO KPP =  2          CKK( 2, 2, 2) =     0.0

ISTDFQ = 100
TIME AND SPACE GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES 21
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL 1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES
   0.0                      0.4290000000000000D+00   0.8590000000000000D+00   0.1288000000000000D+01
   0.1717000000000000D+01   0.2147000000000000D+01   0.2576000000000000D+01   0.3005000000000000D+01
   0.3435000000000000D+01   0.3864000000000000D+01   0.4293000000000000D+01   0.4723000000000000D+01
   0.5152000000000000D+01   0.5581000000000000D+01   0.5841000000000000D+01   0.5900000000000000D+01
   0.6010000000000000D+01   0.6440000000000000D+01   0.6869000000000000D+01   0.7298000000000000D+01
   0.7728000000000000D+01
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  T
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
EPS =    0.1000000000000000D-05      HINIT =    0.1000000000000000D-05      MF =  21      MXGORD =  5
CONTINUITY FOR R AND Z DIRECTIONS MAY HAVE BEEN RESET
CONTR =  3          CONTZ =  0
DEFAULT INITIAL COEFFICIENTS


        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3050844214941653D+03   0.3049508954012962D+03   0.3047766811385551D+03   0.3046431550457051D+03
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3499999992287308D+01   0.3500000011285941D+01   0.3499999997878174D+01   0.3499999984133205D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  2,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3045719980340051D+03   0.3044384719411530D+03   0.3042642576784146D+03   0.3041307315855603D+03
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  2,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3499999983324220D+01   0.3499999996017270D+01   0.3500000020665954D+01   0.3500000027734920D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  3,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3040595745739[illegible]   0.3039260484810054D+03   0.3037518342182685D+03   0.3036183081254161D+03
```

```
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  3,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3500000021935441D+01   0.3499999995090316D+01   0.3499999957284935D+01   0.3499999952004673D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  4,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3035471511137148D+03   0.3034136250208618D+03   0.3032394107581240D+03   0.3031058846652707D+03
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  4,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3499999965274454D+01   0.3500000015534055D+01   0.3500000081371008D+01   0.3500000087287737D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  5,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3030347276535692D+03   0.3029012015607162D+03   0.3027269872979790D+03   0.3025934612051261D+03
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  5,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3500000061348953D+01   0.3499999967545694D+01   0.3499999847188608D+01   0.3499999838362660D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  6,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3025223041934247D+03   0.3023887781005716D+03   0.3022145638378341D+03   0.3020810377449810D+03
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  6,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3499999887560844D+01   0.3500000063001927D+01   0.3500000286157951D+01   0.3500000300206845D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  7,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3020098807332795D+03   0.3018763546404265D+03   0.3017021403776891D+03   0.3015686142848361D+03
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  7,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3500000206708825D+01   0.3499999876642619D+01   0.3499999462905689D+01   0.3499999448187522D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  8,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3014974572731347D+03   0.3013639311802817D+03   0.3011897169175441D+03   0.3010561908246910D+03
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  8,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3499999633451511D+01   0.3500000269277111D+01   0.3500001021777931D+01   0.3500000958966381D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  9,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3009850338129896D+03   0.3008515077201366D+03   0.3006772934573992D+03   0.3005437673645463D+03
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  9,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3500000527133966D+01   0.3499999186632425D+01   0.3499999798488517D+01   0.3499999891649471ZD+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE ( 10,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3004726103528448D+03   0.3003390842599919D+03   0.3001648869972544D+03   0.3000313439C44C14D+03
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE ( 10,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.3500000482538138D+01   0.3500003770169525D+01   0.3499999936380748D+01   0.3499982833460185D+01
```

```
 READING NAMELIST  DATA

 CHANGES IN NAMELIST DATA MAY HAVE BEEN MADE FOR TRANSIENT
FOR SPECIES NO.  1
 SIDE  1 ALPHA =   0.10000000D+01      BETA =   0.0                  GAMMA =   0.1000000D+01
 SIDE  2 ALPHA =   0.0                 BETA =  -0.10000000D+01       GAMMA =   0.0
 SIDE  3 ALPHA =   0.10000000D+01      BETA =   0.0                  GAMMA =   0.1000000D+01
 SIDE  4 ALPHA =   0.0                 BETA =   0.10000000D+01       GAMMA =   0.0
FOR SPECIES NO.  2
 SIDE  1 ALPHA =   0.10000000D+01      BETA =   0.0                  GAMMA =   0.1000000D+01
 SIDE  2 ALPHA =   0.0                 BETA =  -0.10000000D+01       GAMMA =   0.0
 SIDE  3 ALPHA =   0.10000000D+01      BETA =   0.0                  GAMMA =   0.1000000D+01
 SIDE  4 ALPHA =   0.0                 BETA =   0.10000000D+01       GAMMA =   0.0

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  1     NS2=  0     NS3= -1     NS4=  0
 FOR SPECIES NO.  2     NS1= -1     NS2=  0     NS3=  1     NS4=  0
CONVECTION VELOCITY IN R DIRECTION     0.0
CONVECTION VELOCITY IN Z DIRECTION     0.0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
SPECIES NO.  1 MATERIAL INDEX  1 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
SPECIES NO.  2 MATERIAL INDEX  1 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
SPECIES NO.  1 MATERIAL INDEX  1 HU2 =    0.10000000000000000D+01      HU4 =    0.10000000000000000D+01
SPECIES NO.  2 MATERIAL INDEX  1 HU2 =    0.10000000000000000D+01      HU4 =    0.10000000000000000D+01
REACTION RATES
FIRST ORDER RATES
CK INTO  1 FROM  1 IS    0.0
CK INTO  2 FROM  1 IS    0.0
CK INTO  1 FROM  2 IS    0.0
CK INTO  2 FROM  2 IS    0.0
SECOND ORDER REACTION RATES ARE
CKK INTO K =  1 FOR KP =  1 INTO KPP =  1          CKK( 1, 1, 1) =    0.0
CKK INTO K =  1 FOR KP =  2 INTO KPP =  1          CKK( 1, 2, 1) =    0.0
CKK INTO K =  1 FOR KP =  1 INTO KPP =  2          CKK( 2, 1, 1) =    0.0
CKK INTO K =  1 FOR KP =  2 INTO KPP =  2          CKK( 2, 2, 1) =    0.0
CKK INTO K =  2 FOR KP =  1 INTO KPP =  1          CKK( 1, 1, 2) =    0.0
CKK INTO K =  2 FOR KP =  2 INTO KPP =  1          CKK( 1, 2, 2) =    0.0
CKK INTO K =  2 FOR KP =  1 INTO KPP =  2          CKK( 2, 1, 2) =    0.0
CKK INTO K =  2 FOR KP =  2 INTO KPP =  2          CKK( 2, 2, 2) =    0.0

ISTDFQ = 100
TIME AND SPACE GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES 21
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL 1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES
   0.0                    0.42900000000000000D+00   0.85900000000000000D+00   0.12880000000000000D+01
   0.17170000000000000D+01   0.21470000000000000D+01   0.25760000000000000D+01   0.30050000000000000D+01
   0.34350000000000000D+01   0.38640000000000000D+01   0.42930000000000000D+01   0.47230000000000000D+01
   0.51520000000000000D+01   0.55810000000000000D+01   0.58410000000000000D+01   0.59000000000000000D+01
   0.60100000000000000D+01   0.64400000000000000D+01   0.68690000000000000D+01   0.72980000000000000D+01
   0.77280000000000000D+01
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  T
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
EPS =    0.10000000000000000D-05      HINIT =    0.10000000000000000D-05      MF =  21      MXORD =  5
CONTINUITY FOR R AND Z DIRECTIONS
CONTR =  3          CONTZ =  0
```

INITIAL COEFFICIENTS FOR TRANSIENT

0.3051200000000230D+03	0.3499999979998900D+01	0.3049491921799380D+03	0.3500000047755501D+01
0.3046075765398622D+03	0.3499999922026616D+01	0.3040951530797070D+03	0.3500000098689985D+01
0.3035827296195667D+03	0.3499999838219784D+01	0.3030703061594194D+03	0.3500000290486270D+01
0.3025578826992759D+03	0.3499999463305600D+01	0.3020454592391300D+03	0.3500001001598519D+01
0.3015330357789858D+03	0.3499998111794399D+01	0.3010206123188402D+03	0.3500003680296129D+01
0.3005081888586958D+03	0.3499991861774387D+01	0.3001665732185990D+03	0.3500014932673946D+01
0.2999957653985508D+03	0.3499975298071043D+01		

```
 BEGIN TRANSIENT SOLUTION


PROUT FOR TIME =        0.0

  W =
      0.30512000D+03      0.35000000D+01      0.30494919D+03      0.35000000D+01      0.30460758D+03      0.34999999D+01
      0.30409515D+03      0.35000001D+01      0.30358273D+03      0.34999998D+01      0.30307031D+03      0.35000003D+01
      0.30255788D+03      0.34999995D+01      0.30204546D+03      0.35000010D+01      0.30153304D+03      0.34999981D+01
      0.30102061D+03      0.35000037D+01      0.30050819D+03      0.34999919D+01      0.30016657D+03      0.35000149D+01
      0.29999577D+03      0.34999753D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.30512000000000229D+03   0.30409515307970940+03   0.30307030615942000+03   0.30204545923913030+03   0.30102061231884030+03
   0.29999576539855070+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.34999999799989000+01   0.35000000258343890+01   0.35000000772450770+01   0.35000002635823450+01   0.35000007824588830+01
   0.34999752980710430+01
   TOUT=   0.0


PROUT FOR TIME =        0.42900000D+00

  W =
      0.30512000D+03      0.34997975D+01      0.30490589D+03      0.35010219D+01      0.30468000D+03      0.34986471D+01
      0.30402964D+03      0.35007032D+01      0.30364543D+03      0.35001866D+01      0.30300766D+03      0.34986612D+01
      0.30268892D+03      0.35021127D+01      0.30153048D+03      0.35000112D+01      0.30594146D+03      0.34685926D+01
      0.31073420D+03      0.34180421D+01      0.31377625D+03      0.33920378D+01      0.31812047D+03      0.33600726D+01
      0.31872882D+03      0.33452813D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.30511999999999980+03   0.30407401206642020+03   0.30306083180262290+03   0.30245871557058330+03   0.31044241781087790+03
   0.31872882160339610+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.34997974857713270+01   0.35002743912047440+01   0.34994906741563680+01   0.34951249974159170+01   0.34221331556969820+01
   0.33452812840300080+01
   TOUT=   0.42900000000000000+00  DELTA T =   0.42900000000000000+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.43601399D+03
TIME LEFT       0.44706000D+05


PROUT FOR TIME =        0.85900000D+00

  W =
      0.30512000D+03      0.34995947D+01      0.30496867D+03      0.34966589D+01      0.30437124D+03      0.35051747D+01
      0.30407290D+03      0.34989284D+01      0.30751750D+03      0.34674934D+01      0.31302244D+03      0.34216478D+01
      0.31590380D+03      0.33873741D+01      0.32128564D+03      0.33498509D+01      0.32533242D+03      0.33013590D+01
      0.32958951D+03      0.32724280D+01      0.33535824D+03      0.32186584D+01      0.33725844D+03      0.31946876D+01
```

```
      0.33953476D+03      0.31913697D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999998D+03   0.3046967203375276D+03   0.3125851763844824D+03   0.3210631310840153D+03   0.3298414488540445D+03
   0.3395347598578759D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3499594711359489D+01   0.3494730278803098D+01   0.3423576481046234D+01   0.3348022804567568D+01   0.3268288241072877D+01
   0.3181369709858518D+01
   TOUT=   0.9590000000000000D+00  DELTA T =   0.4300000000000000D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.15663488D+03
TIME LEFT       0.44549000D+05


PROUT FOR TIME =       0.12880000D+01

  W =
      0.3051200000+03      0.33484102D+01      0.30810081D+03      0.33501745D+01      0.31418574D+03      0.33535068D+01
      0.3229570400+03      0.33508902D+01      0.32767759D+03      0.33076997D+01      0.33135609D+03      0.32666271D+01
      0.3368899900+03      0.32277283D+01      0.34103714D+03      0.31774192D+01      0.34615608D+03      0.31472262D+01
      0.3513936200+03      0.30863322D+01      0.35578983D+03      0.30562490D+01      0.36003677D+03      0.30198036D+01
      0.3614751800+03      0.30031981D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999999D+03   0.3222819145036616D+03   0.3316643232829942D+03   0.3411981063971791D+03   0.3512533952052090D+03
   0.3614751794530537D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3344941021927506D+01   0.3344127869502941D+01   0.3266989422656087D+01   0.3180771879859237D+01   0.3091467316282912D+01
   0.3003198099497755D+01
   TOUT=   0.1288000000000000D+01  DELTA T =   0.4290000000000000D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.14100000D+03
TIME LEFT       0.44408000D+05


PROUT FOR TIME =       0.17170000D+01

  W =
      0.3051200000+03      0.30384720D+01      0.3083246CD+03      0.30356893D+01      0.314925CCD+03      0.30365009D+01
      0.3242295200+03      0.30311295D+01      0.33456699D+03      0.30334143D+01      0.343280C4D+03      0.30179598D+01
      0.3540431100+03      0.30183550D+01      0.36327759D+03      0.30116741D+01      0.36887720D+03      0.29523359D+01
      0.3738693600+03      0.29189712D+01      0.37956791D+03      0.28565369D+01      0.38357124D+03      0.28282452D+01
      0.3853553900+03      0.28123782D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999999D+03   0.3244010094392673D+03   0.3436217097572645D+03   0.3626711755695567D+03   0.3739870944241103D+03
   0.3853553997815424D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
```

```
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3038471963006792D+01   0.3032405512722226D+01   0.3020601385500648D+01   0.3002897857756941D+01   0.2914126270492313D+01
   0.2812378172850081D+01
   TOUT=   0.1717000000000000D+01  DELTA T =   0.4290000000000000D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.11827506D+03
TIME LEFT       0.44290000D+05



PROUT FOR TIME =       0.21470000D+01

  W =
      0.30512000D+03      0.26951713D+01      0.30863695D+03      0.26970174D+01      0.31531739D+03      0.26962833D+01
      0.32659718D+03      0.26951396D+01      0.33619348D+03      0.26847761D+01      0.34790844D+03      0.26867501D+01
      0.35717209D+03      0.26722768D+01      0.36887965D+03      0.26619116D+01      0.37905812D+03      0.26603204D+01
      0.38900891D+03      0.26287781D+01      0.40154248D+03      0.26313309D+01      0.40724960D+03      0.26144062D+01
      0.41024526D+03      0.26034897D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999998D+03   0.3263166008477955D+03   0.3474998879323223D+03   0.3686247988293743D+03   0.3894393752629132D+03
   0.4102452639767139D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.2695171270487139D+01   0.2693602941899904D+01   0.2684008838514435D+01   0.2663373899677314D+01   0.2634460641706568D+01
   0.2603489663279575D+01
   TOUT=   0.2147000000000000D+01  DELTA T =   0.4299999999999999D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.13967442D+03
TIME LEFT       0.44150000D+05



PROUT FOR TIME =       0.25760000D+01

  W =
      0.30512000D+03      0.23326113D+01      0.30888527D+03      0.23338313D+01      0.31675592D+03      0.23279968D+01
      0.32766501D+03      0.23277249D+01      0.33996315D+03      0.23264328D+01      0.35027202D+03      0.23125945D+01
      0.36319586D+03      0.23094137D+01      0.37343671D+03      0.22967026D+01      0.38376939D+03      0.22953085D+01
      0.39324124D+03      0.23104686D+01      0.40082462D+03      0.23021087D+01      0.40918074D+03      0.23061698D+01
      0.41115667D+03      0.23084489D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999999D+03   0.3278965156587021D+03   0.3507078463373167D+03   0.3734520128926559D+03   0.3929264920060352D+03
   0.4111566745952916D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.2332611322758141D+01   0.2327554844887493D+01   0.2314370734257793D+01   0.2298588749976574D+01   0.2306548618184080D+01
   0.2308448916163730D+01
   TOUT=   0.2576000000000000D+01  DELTA T =   0.4290000000000000D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.10200000D+03
```

```
TIME LEFT       0.44048000D+05


PROUT FOR TIME =       0.30050000D+01

  W =
      0.30512000D+03       0.19382478D+01       0.30937373D+03       0.19335977D+01       0.31761221D+03       0.19420041D+01
      0.33010012D+03       0.19322842D+01       0.34040375D+03       0.19412603D+01       0.35101292D+03       0.19609385D+01
      0.36014243D+03       0.19612835D+01       0.37144820D+03       0.19753239D+01       0.38040357D+03       0.19837370D+01
      0.38971214D+03       0.19894355D+01       0.40105180D+03       0.19988751D+01       0.40485645D+03       0.20037422D+01
      0.40906959D+03       0.20054067D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.30511999999999999D+03   0.32973607566410600+03   0.35076630812735660+03   0.37105646808387710+03   0.39005065629335080+03
   0.40906959204021640+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.19382477868185030+01   0.19354002262518200+01   0.19577163259653050+01   0.19743860544583760+01   0.19900590286464230+01
   0.20054067299863510+01
   TOUT=   0.30050000000000000+01  DELTA T =   0.42899999999999980+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.10223776D+03
TIME LEFT       0.43946000D+05


PROUT FOR TIME =       0.34350000D+01

  W =
      0.30512000D+03       0.15807566D+01       0.30790675D+03       0.15891471D+01       0.31372724D+03       0.15705055D+01
      0.32329415D+03       0.15846681D+01       0.33367367D+03       0.15977550D+01       0.34412057D+03       0.16115938D+01
      0.35556590D+03       0.16321241D+01       0.36441688D+03       0.16408396D+01       0.37541837D+03       0.16601420D+01
      0.38549656D+03       0.16690684D+01       0.39361964D+03       0.16853975D+01       0.40190825D+03       0.16927177D+01
      0.40415220D+03       0.16972465D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.30511999999999990+03   0.32342958216732090+03   0.34428697299333920+03   0.36477529634394560+03   0.38517070411141120+03
   0.40415219821131940+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.15807566268352250+01   0.15944888111218340+01   0.16127090413618310+01   0.16426040938343040+01   0.16703022152638290+01
   0.16972464666581220+01
   TOUT=   0.34350000000000000+01  DELTA T =   0.42000000000020020+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.91786047D+02
TIME LEFT       0.43853000D+05


PROUT FOR TIME =       0.38640000D+01

  W =
      0.30512000D+03       0.12936218D+01       0.30819747D+03       0.12850341D+01       0.31418145D+03       0.13007441D+01
```

```
      0.32173020D+03      0.12887516D+01      0.33117282D+03      0.12964629D+01      0.33992789D+03      0.13016910D+01
      0.34722796D+03      0.12962514D+01      0.35717637D+03      0.13120074D+01      0.3662805CD+03      0.13212454D+01
      0.37668319D+03      0.13534177D+01      0.38731946D+03      0.13647554D+01      0.392534C8D+C3      0.13809226D+01
      0.39608701D+03      0.13878047D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.305119999999999D+03   0.3220458470221918D+03   0.339695389569081 8D+03   0.3570356559552698D+03   0.3767221200403455D+03
   0.3960870119262076D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.1293621816526897D+01   0.1292035563453516D+01   0.1299913068963711D+01   0.1310921062237475D+01   0.1349945261929964D+01
   0.1387804658954566D+01
   TOUT=   0.3864000000000000D+01  DELTA T =   0.4290000000000000D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.940000000+02
TIME LEFT       0.43759000D+05


PROUT FOR TIME =       0.42930000D+01

  W =
      0.30512000D+03      0.10158056D+01      0.30766970D+03      0.10223020D+01      0.3127269CD+03      0.10120605D+01
      0.32220821D+03      0.10239186D+01      0.32881565D+03      0.10211831D+01      C.33761264D+03      0.102591C2D+01
      0.34594218D+03      0.10366839D+01      0.35280078D+03      0.10318060D+01      0.3618C733D+03      0.10543581D+01
      0.36920031D+03      0.10504283D+01      0.37630815D+03      0.10699975D+01      0.381768C9D+03      0.10751828D+01
      0.38498554D+03      0.10799251D+01
 VALUES CF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.305119999999999D+03   0.3217292343213048D+03   0.3375347325019273D+03   0.3531587689430451D+03   0.3691527888094773D+03
   0.3849855406251316D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.1015805572152347D+01   0.1021486368508123D+01   0.1026917946220128D+01   0.1036377687561939D+01   0.1054344803443059D+01
   0.1079925133842168D+01
   TOUT=   0.4293000000000000D+01  DELTA T =   0.4289999999999998D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.10324009D+03
TIME LEFT       0.43656000D+05


PROUT FOR TIME =       0.47230000D+01

  W =
      0.30512000D+03      0.76387437D+00      0.30775050D+03      0.76076058D+00      0.31331583D+03      0.76566250D+00
      0.31893419D+03      0.76331880D+00      0.32834289D+03      0.77096881D+00      0.33417569D+03      0.77464160D+00
      0.34200318D+03      0.78122990D+00      0.34954447D+03      0.79779002D+00      0.35715934D+03      0.78499709D+00
      0.36684753D+03      0.79650654D+00      0.37499818D+03      0.78597958D+00      0.38107527D+03      0.78916798D+00
      0.38361475D+03      0.78954939D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
```

```
   0.30511999999999990+03   0.31956591114457540+03   0.33450813730373040+03   0.34955673052958850+03   0.36659127144226770+03
   0.38361475389656980+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.76387437175186990+00   0.76499442163082920+00   0.77512752003332940+00   0.79289784306383330+00   0.79283380624061940+00
   0.78954939024486470+00
   TOUT=   0.47230000000000000+01  DELTA T =   0.43000000000000020+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.92783721D+02
TIME LEFT       0.43563000D+05


PROUT FOR TIME =       0.51520000D+01

  U =
      0.30512000D+03      0.53395729D+00      0.30714189D+03      0.53324276D+00      0.31070400D+03      0.53777800D+00
      0.31927701D+03      0.53892685D+00      0.32463987D+03      0.53232250D+00      0.33500939D+03      0.52816738D+00
      0.34182616D+03      0.52146924D+00      0.35033662D+03      0.51608373D+00      0.35914949D+03      0.51778540D+00
      0.36693769D+03      0.50395205D+00      0.37648738D+03      0.51015748D+00      0.38152087D+03      0.50497264D+00
      0.38505810D+03      0.50271314D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.30511999999999990+03   0.31874198128340830+03   0.33441726461226200+03   0.35038702238436720+03   0.36723127249946990+03
   0.38505810255409780+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.53395729140501720+00   0.53763465396474520+00   0.52774354355401890+00   0.51726492494902910+00   0.50729184487020130+00
   0.50271313573172510+00
   TOUT=   0.51520000000000000+01  DELTA T =   0.42899999999999980+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.97000000D+02
TIME LEFT       0.43466000D+05


PROUT FOR TIME =       0.55810000D+01

  U =
      0.30512000D+03      0.29181934D+00      0.30819657D+03      0.29502858D+00      0.31479989D+03      0.28842729D+00
      0.32144462D+03      0.29306101D+00      0.33126395D+03      0.28287176D+00      0.33691902D+03      0.26405535D+00
      0.34602298D+03      0.26363316D+00      0.35357948D+03      0.24760937D+00      0.36176713D+03      0.23946724D+00
      0.37067840D+03      0.23528125D+00      0.37825202D+03      0.22206476D+00      0.38525881D+03      0.21695265D+00
      0.38753790D+03      0.21508157D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.30511999999999990+03   0.32197372109680380+03   0.33749383368520630+03   0.35368467472511420+03   0.37045546017587190+03
   0.38753790169121850+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.29181934317468450+00   0.29059051706984810+00   0.26712105612756720+00   0.24892231027655220+00   0.23377616598155240+00
   0.21508156989094050+00
```

```
   TOUT=   0.5581000000000000D+01  DELTA T =   0.4290000000000000D+00

ESTIMATED TIME FOR A CALL TO DRIVE        0.56363636D+02
TIME LEFT       0.43373000D+05


PROUT FOR TIME =        0.58410000D+01

  W =
      0.30512000D+03      0.1186204 0D+00      0.30812423D+03      0.12826822D+00      0.3142536CD+03      0.10756583D+00
      0.32257718D+03      0.12080672D+00      0.33209857D+03      0.11832817D+00      0.34C84C5CD+03      0.10880187D+00
      0.34947846D+03      0.11225439D+00      0.35671307D+03      0.85875126D-01      0.36504950D+03      0.80738436D-01
      0.37376418D+03      0.63916084D-01      0.38130220D+03      0.53272248D-01      0.38850912D+03      0.43313060D-01
      0.39078890D+03      0.39946500D-01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999998D+03   0.3227768177585234D+03   0.3408231726745876D+03   0.3568967100235479D+03   0.3735680733850098D+03
   0.3907889013177352D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.1186204003227742D+00   0.1181868151915035D+00   0.1109650083616525D+00   0.8941555539886140D-01   0.6494583624731928D-01
   0.3994649974351334D-01
   TOUT=   0.5841000000000000D+01  DELTA T =   0.2600000000000000D+00

ESTIMATED TIME FOR A CALL TO DRIVE        0.13388462D+02
TIME LEFT       0.43314000D+05


PROUT FOR TIME =        0.59000000D+01

  W =
      0.30512000D+03      0.79115611D-01      0.30829769D+03      0.79946586D-01      0.3150C1731D+03      0.77657022D-01
      0.32172128D+03      0.77833234D-01      0.33303651D+03      0.75622287D-01      0.34C5419CD+03      0.72856414D-01
      0.35019607D+03      0.70989894D-01      0.35815939D+03      0.56989157D-01      0.365522C0D+03      0.37983135D-01
      0.37484446D+03      0.31000172D-01      0.38226809D+03      0.10950803D-01      0.3893390 1D+03      0.38258330D-02
      0.39148360D+03      0.20202355D-04
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999999D+03   0.3224898236391072D+03   0.3409000269270767D+03   0.3580592719C4E873D+03   0.3745279883209404D+03
   0.3914836003118876D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.7911561077177055D-01   0.7743537433945341D-01   0.7300630588092091D-01   0.5615494287938823D-01   0.2882243799326704D-01
   0.2020235499551192D-04
   TOUT=   0.5900000000000000D+01  DELTA T =   0.5899999999999994D-01

ESTIMATED TIME FOR A CALL TO DRIVE        0.19203390D+03
TIME LEFT       0.43210000D+05


PROUT FOR TIME =        0.60100000D+01
```

```
  W =
      0.30512000D+03      0.43030812D-02      0.30804722D+03     -0.34158102D-02      0.31359433D+03      0.12679068D-01
      0.32403527D+03     -0.12631084D-05      0.33159701D+03     -0.26233500D-02      0.34179164D+03      0.27780114D-02
      0.35035217D+03     -0.10930790D-01      0.36005268D+03     -0.58692050D-02      0.3678727CD+03     -0.30559532D-01
      0.37598473D+03     -0.37283151D-01      0.38566963D+03     -0.75892938D-01      0.385462C9D+03     -0.10602568D-01
      0.38309902D+03      0.0
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999999D+03   0.3235554014593303D+03   0.3415192887166297D+03   0.3597392656308896D+03   0.3762468780379001D+03
   0.3830990196265668D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.4003081159989102D-02   0.1675110907339278D-02  -0.4070158013325739D-03  -0.1082785707414835D-01  -0.4259584575317032D-01
   0.0
   TOUT=   0.6010000000000000D+01  DELTA T =   0.1100000000000001D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.91863636D+03
TIME LEFT       0.42975000D+05


PROUT FOR TIME =       0.64400000D+01

  W =
      0.30512000D+03     -0.29801095D+00      0.30823027D+03     -0.28590567D+00      0.31474581D+03     -0.31280854D+00
      0.32416768D+03     -0.29551106D+00      0.33379258D+03     -0.29938609D+00      0.34383479D+03     -0.31936719D+00
      0.35352964D+03     -0.30684553D+00      0.35465189D+03     -0.26074420D+00      0.35386375D+03     -0.17134942D+00
      0.35684110D+03     -0.12718341D+00      0.35653451D+03     -0.66050139D-01      0.35764976D+03     -0.17310502D-02
      0.35944244D+03      0.0
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999999D+03   0.3242015201424632D+03   0.3437768996880991D+03   0.3543334927209423D+03   0.3562937788131229D+03
   0.3594424436102679D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
  -0.2980109549035217D+00  -0.2990398086111563D+00  -0.3139500625055814D+00  -0.2535286220663324D+00  -0.1243555315579401D+00
   0.0
   TOUT=   0.6440000000000000D+01  DELTA T =   0.4299999999999999D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.21649535D+03
TIME LEFT       0.42758000D+05


PROUT FOR TIME =       0.68690000D+01

  W =
      0.30512000D+03     -0.62394871D+00      0.30891116D+03     -0.63937277D+00      0.31600776D+03     -0.60147008D+00
      0.31702958D+03     -0.56538482D+00      0.31774684D+03     -0.47475577D+00      0.32002481D+03     -0.41210136D+00
      0.32115018D+03     -0.36226831D+00      0.32295804D+03     -0.27616171D+00      0.32322614D+03     -0.22920468D+00
      0.32338069D+03     -0.13724414D+00      0.32358610D+03     -0.66384964D-01      0.32218047D+03     -0.40756950D-01
      0.32556991D+03      0.0
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
```

```
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999999D+03   0.3169788161520082D+03   0.3198327102971742D+03   0.3227014104978888D+03   0.3233891629508882D+03
   0.3255699107527535D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
  -0.6239487068946069D+00  -0.5562941910757348D+00  -0.4142382536873556D+00  -0.2826866368542663D+00  -0.1407610332609688D+00
   0.0
   TOUT=   0.6869000000000000D+01  DELTA T =   0.4290000000000000D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.18600000D+03
TIME LEFT       0.42571000D+05


PROUT FOR TIME =       0.72980000D+01

  W =
      0.30512000D+03     -0.53800556D+00      0.30322814D+03     -0.52424068D+00      0.29877293D+03     -0.54447069D+00
      0.29218445D+03     -0.53472208D+00      0.28344993D+03     -0.53887977D+00      0.28494027D+03     -0.46235098D+00
      0.28575632D+03     -0.36756478D+00      0.28607818D+03     -0.31566002D+00      0.28754874D+03     -0.21141730D+00
      0.28601946D+03     -0.16587872D+00      0.28844543D+03     -0.57589512D-01      0.28783909D+03     -0.38033445D-01
      0.28538311D+03      0.0
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999999D+03   0.2918267756698008D+03   0.2848278896340909D+03   0.2862696294762404D+03   0.2866786709408834D+03
   0.2853831071152571D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
  -0.5380055557433031D+00  -0.5370397971333473D+00  -0.4593080900375815D+00  -0.3069370251372912D+00  -0.1554202810198169D+00
   0.0
   TOUT=   0.7298000000000000D+01  DELTA T =   0.4290000000000000D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.20247086D+03
TIME LEFT       0.42369000D+05


PROUT FOR TIME =       0.77280000D+01

  W =
      0.30512000D+03     -0.29035684D+00      0.30199187D+03     -0.29444264D+00      0.29706097D+03     -0.28823124D+00
      0.28655730D+03     -0.29265326D+00      0.28157238D+03     -0.27973189D+00      0.27081172D+03     -0.26906973D+00
      0.26652020D+03     -0.25620446D+00      0.25530049D+03     -0.24935938D+00      0.24803424D+03     -0.25119788D+00
      0.24905795D+03     -0.15191249D+00      0.24743394D+03     -0.10007929D+00      0.25139421D+03     -0.59618956D-02
      0.24741893D+03      0.0
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3051199999999998D+03   0.2874770896301570D+03   0.2718899114706059D+03   0.2559593975587316D+03   0.2486166611812982D+03
   0.2474189309348203D+03
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )
```

```
APPROXIMATE SOLUTION
  -0.2903568438053339D+00  -0.2897626966755584D+00  -0.2687025442579254D+00  -0.2508066439447891D+00  -0.1598211874396352D+00
   0.0
   TOUT=    0.7728000000000000D+01  DELTA T =    0.4299999999999999D+00

ESTIMATED TIME FOR A CALL TO DRIVE       0.2070000CD+03
TIME LEFT       0.42162000D+05

****** NORMAL DUMP AT END OF TIMEX
         I =    21          TOUT =    0.7728000000C0C0C0D+01


END OF CASE
```

We next illustrate the use of the cross-section plot (CSP) program to generate a series of graphs at selected times. Much of the input is similar to that given in section 7.1. For example, the graphics dataset which was written on unit 12 during the computational phase must be reassigned to unit 12 for the CSP run.

For CSP, two namelists are required. The first is Namelist FØRMAT which has input very similar to the FØRMAT namelist used in section 7.1. The second is Namelist CSPIN which involves variables specifically for the CSP program.

Namelist FØRMAT

1. Indicator for iterative or direct version of the computational code

   The direct version was used and so
   ITRTV=0, (Default)

2. Number of curves to be produced per time value

   For this problem we will generate two curves (one for each species) on each frame for each time value. Each curve will have its own axis.
   IGNUM=2,

3. Number of grid points for graphical purposes

   This number cannot exceed NRES.
   NRESIN=NRES1, (NRES1=501)

We next consider the namelist which specifically applies to the CSP program.

Namelist CSPIN

1. Indicator for cinema mode

   We would use this option only if we wished to produce multiple copies of each frame. This should only be done in movie generation.
   ICN=0, (Default)

2. Indicator for grouping format

When more than one curve appears on a single frame, we can either produce the curves on a single set of axes (packed format) or produce each curve on its own set of axes (separate format). Since the range of values for pressure and velocity differ considerably, it is not desirable to use packed mode for this problem. We thus use the separate mode.
IFØRMT=0, (Default)

3. Indicator for analytic solution

If the analytic solution is available we can graph the analytic solution for the Ith species with its corresponding numerical solution. The analytic solution is supplied via a SINGLE PRECISION version of the analytic subroutine. Since the CSP program contains its own dummy version of the analytic subroutine we don't need to provide the dummy version. For this problem we do not have an analytic solution; we use this dummy routine and set
IANAL=F, (Default)

4. Estimate of minimum value of the ordinates

This vector has, as its Ith component, an estimate for the lower bound of the vertical axis for the Ith curve. When the solution values lie outside the interval [YAXMIN(I),YAXMAX(I)], the code will rescale the vertical axis. Hence if the user does not want this rescaling to occur, he should specify a generous interval which contains a vertical axis range large enough for the Ith curve over all time values being plotted. Also, such a constant axis is useful for movie generation. In other cases it is sufficient to use the default values.
YAXMIN=0.0,0.0, (Default)

5. Estimate of maximum value of the ordinates

YAXMAX=1.0,1.0, (Default)

6. Species number for each curve

This vector has, as its Ith component, the species to be plotted as

the Ith curve. In the usual case, which we have, we use ISPEC(I)=I, for I=1,NSPEC.
ISPEC=1,2,

7. Frame number indicator

This vector specifies the frame on which the Ith curve is to be plotted. In our case we are plotting one frame for each time value.
LGRØUP=1,1, (Default)

8. Ordering of the curves on each frame

If IFØRMT=1, the Ith component of this vector specifies the plotting symbol to be used to indicate the Ith curve. If IFØRMT≠1 (as in this case), LØRDER(I) indicates the order of the curves on each frame.
LØRDER=1,2,

9. R coordinate of first endpoint of the cross-section line

This vector has, as its mth component, the R coordinate of the first of the two points specifying the cross-section for the mth frame. That is, each frame can have a different cross-section if desired. In our case we have only one frame and we wish to plot along the R axis from 0.0 to 4253.5. Further, since the Z axis defaults to [0.0,1.0], we can use any constant in the interval [0.0,1.0] for the Z component. We will use a cross-section from [0.0,0.0] to [4253.5,0.0].
A1=0.0, (Default)

10. Z coordinate of first endpoint of the cross-section

B1=0.0, (Default)

11. R coordinate of second endpoint of the cross-section

A2=4253.5,

12. Z coordinate of second endpoint in cross-section

B2=0.0, (Default)

The following page contains the printed output of the CSP run. This is followed by three of the twenty-one frames generated by the run. Each frame is for a given time value and consists of two curves: the lower curve (species 1) is a plot of head pressure versus distance; the upper curve (species 2) is a plot of velocity versus distance. Figure 7.2.1 is at a time (t = 3.435) before the valve is closed. Figure 7.2.2 is at the time (t = 5.9) when the valve has just closed, and Figure 7.2.3 is at a time (t = 7.728) after the valve has closed. Note the flow reversal in this last figure.

BEGINNING CROSS SECTION GRAPHICS PACKAGE

FORMATTING PARAMETERS

CINEMA MODE	VERSION	NUMBER OF GRAPHS FOR EACH TIME	GROUPING FORMAT	ANALYTIC
0	2	2	0	F

NRESIN = 501

ITIME = F NUMBER OF TIME VALUES FOR TIME OPTION IS 501
IF ITIME=T, THE SOLUTION IS EVAIUATED AT (A1(I),B1(I))

INITIAL ESTIMATE FOR VERTICAL AXES
YAXMIN,YAXMAX
0.0 0.0

USER GRAPH FORMAT SPECIFICATIONS

FRAME NUMBER	ORDER WITHIN EACH FRAME FROM TOP TO BOTTOM	SPECIES NUMBER
1	1	1
	2	2

COORDINATES OF THE ENDPOINTS FOR EACH CROSS SECTION LINE GIVEN WITH THE CORRESPONDING FRAME NUMBER

FRAME NUMBER	X1	Y1	X2	Y2
1	0.0	0.0	0.42535000E+04	0.0

THE END OF THE DATA HAS BEEN REACHED, PROGRAM ENDS.

THE NUMBER OF GRAPHS PRODUCED IS 21

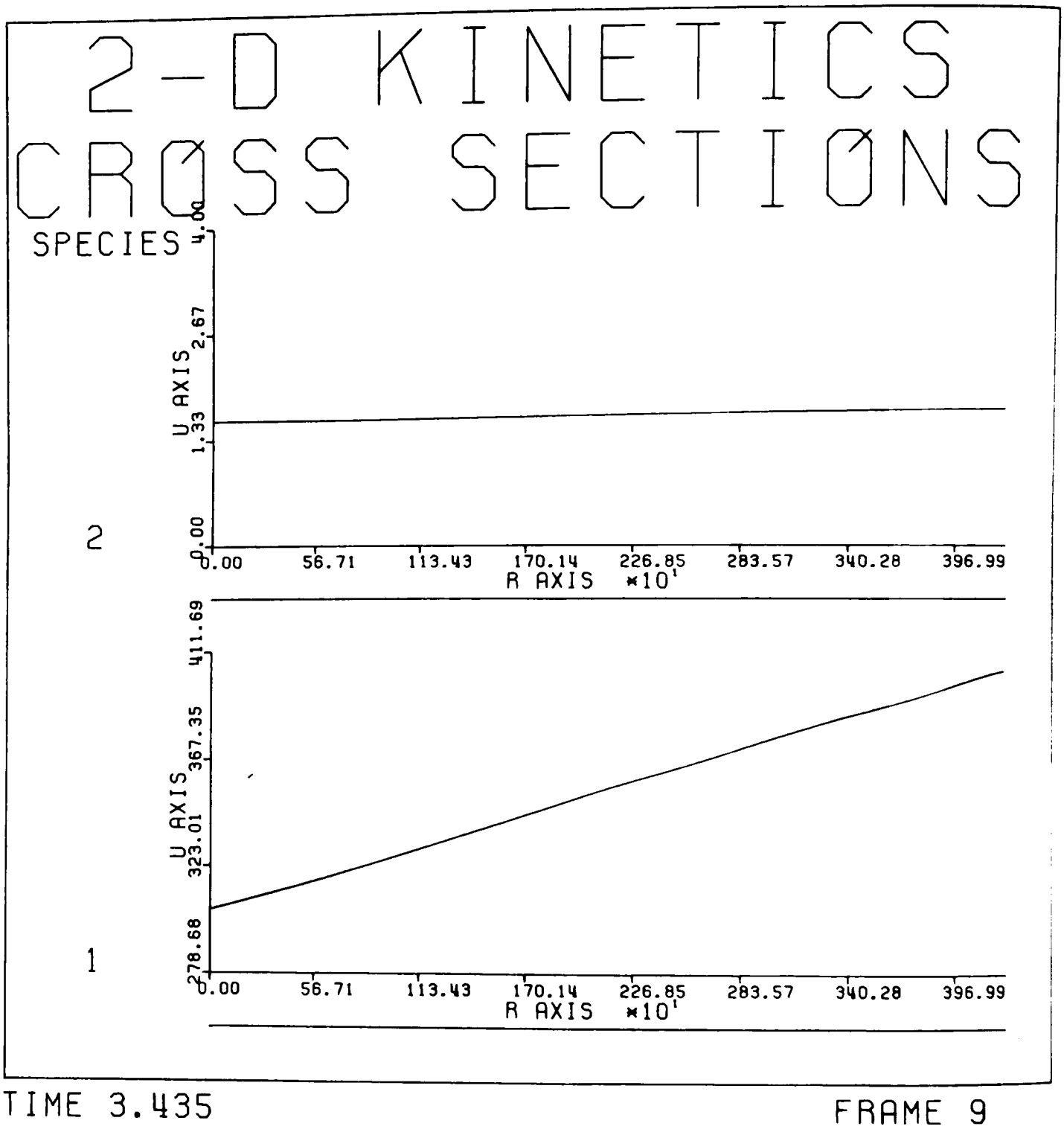


Figure 7.2.1
Graph of pressure and velocity for water hammer problem

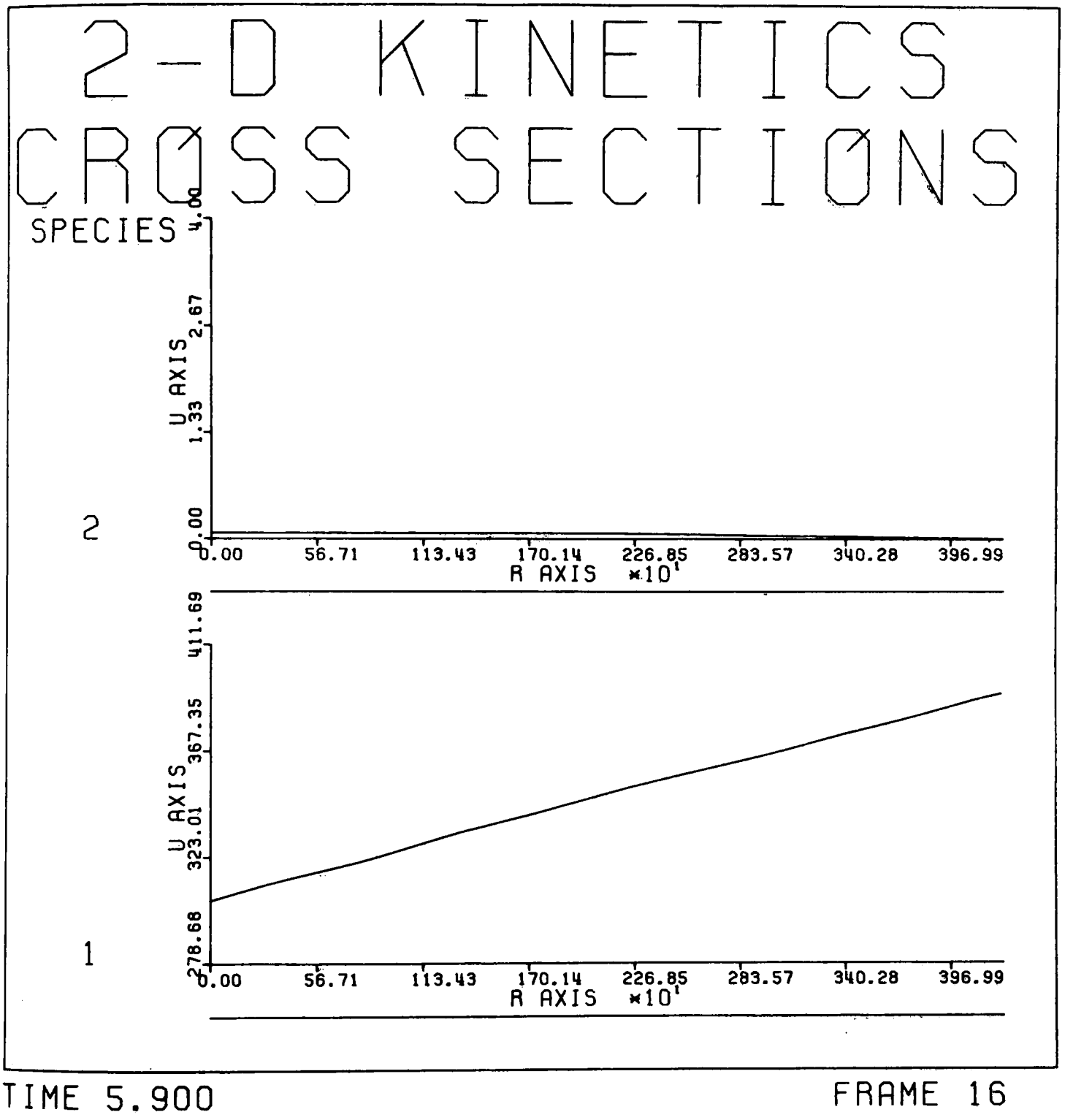


Figure 7.2.2

Graph of pressure and velocity for water hammer problem

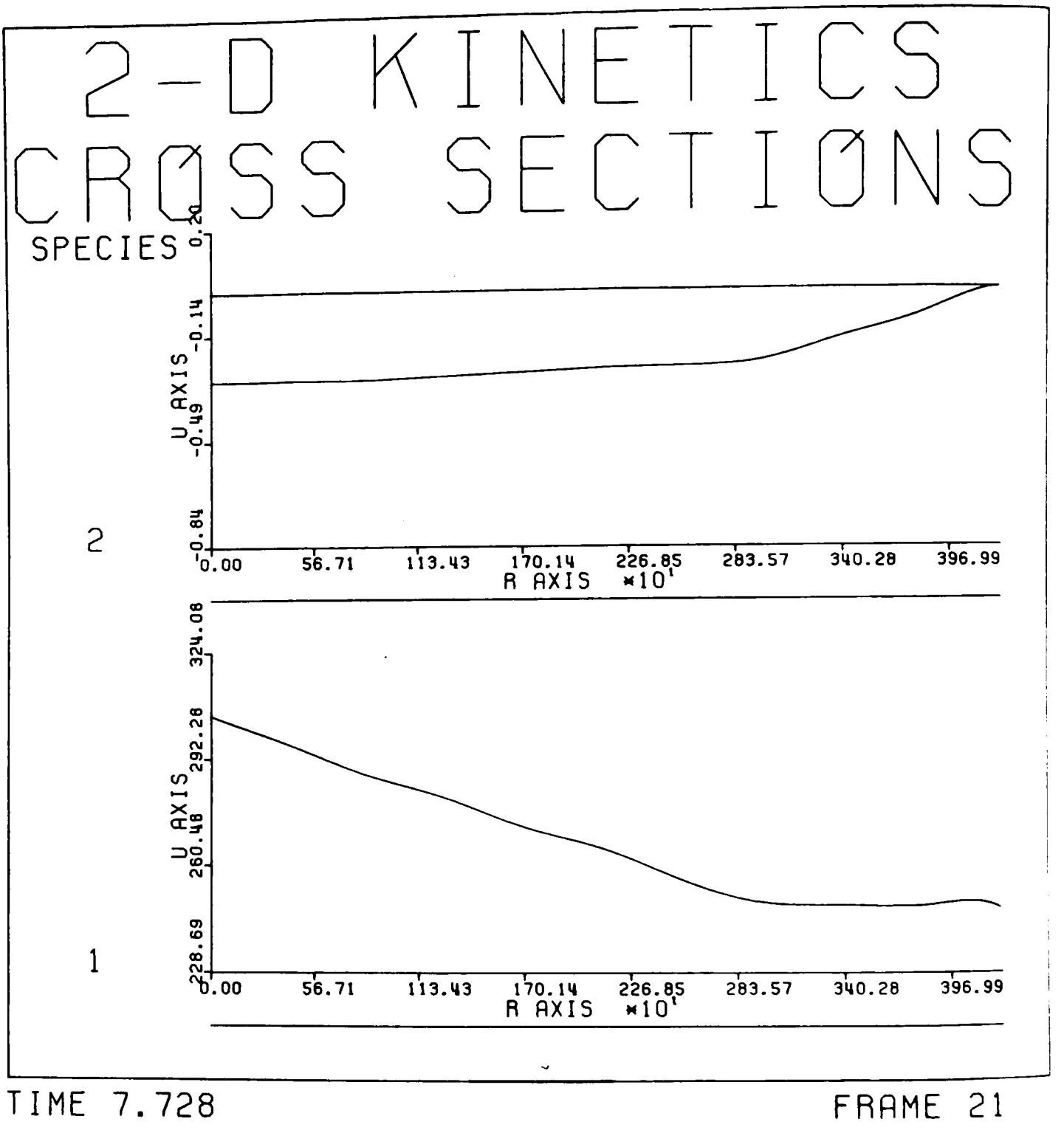


Figure 7.2.3
Graph of pressure and velocity for water hammer problem

7.3 Diurnal Chemical Kinetics

This problem is concerned with the concentrations of minor chemical species in the upper atmosphere. A major feature of this problem is the large time variations in the concentrations of the chemical species. In the course of solving this problem we shall see how a knowledgeable user can modify the code (at least its performance) by modifying the user subroutines. This problem will also illustrate the time history graphical output.

In the following problem, we let $c^1(z,t)$ denote the concentration of ozone (O_3), $c^2(z,t)$ denote the concentration of the oxygen singlet (0), and c^3 denote the concentration of oxygen (O_2) assumed to be constant. This model neglects convection, uses one spatial coordinate z (the altitude in kilometers), and a Fickian model of turbulent eddy diffusion. The problem is as follows.

$$\frac{\partial c^i}{\partial t} = \frac{\partial}{\partial z}\left[K(z)\,\frac{\partial c^i}{\partial z}\right] + R^i(c,t), \quad i=1,2$$

$$\frac{\partial c^i(30,t)}{\partial z} = \frac{\partial c^i(50,t)}{\partial z} = 0, \quad t > 0$$

$$c = (c^1(z,t),\ c^2(z,t))^T,$$

$$30 \le z \le 50,$$

$$0 \le t \le 8.64\cdot 10^4 \text{ (one 24-hour day in seconds).}$$

Further,

$$K(z) = \exp[z/5],$$

$$R^1(c,t) = -k_1 c^1 c^3 - k_2 c^1 c^2 + 2k_3(t)c^3 + k_4(t)c^2$$

$$R^2(c,t) = k_1 c^1 c^3 - k_2 c^1 c^2 - k_4(t)c^2$$

$$c^3 = 3.7\cdot 10^{16}$$

$$k_1 = 1.63\cdot 10^{-16}$$

$$k_2 = 4.66\cdot 10^{-16}$$

$$k_i(t) = \begin{cases} \exp(-\nu_i/\sin \omega t), & \text{for } \sin \omega t > 0, \\ 0, & \text{for } \sin \omega t \leq 0, \end{cases}$$

$$\nu_3 = 22.62, \quad \nu_4 = 7.601, \text{ and } \quad \omega = \pi/43{,}200.$$

$$c^1(z,0) = 10^6\gamma(z), \quad c^2(z,0) = 10^{12}\gamma(z), \text{ where}$$

$$\gamma(z) = 1 - \left(\frac{z-40}{10}\right)^2 + \frac{1}{2}\left(\frac{z-40}{10}\right)^4.$$

Notice that the reaction rates $k_3(t)$ and $k_4(t)$ build up to a peak at noon ($t = 21{,}600$) and are switched off from sunset ($t = 43{,}200$) to sunrise ($t = 86{,}400$) which models the diurnal effect. In this model the concentration $c^1(z,t)$ rises to a peak value of about 10^8 at noon, and then falls to zero at 6:00 p.m. and stays zero through the night. This behavior requires a modification to the error control mechanism in the ODE solver GEAR. Such a modification could have been made in GEAR; however, we choose to make this adjustment through a user routine in order to illustrate the point that some modifications can be achieved through these user routines. Let Y(I) denote the I-th component of the solution vector and E(I) the I-th component of the error vector, both considered at some current time t. The version of GEAR which is used in this code defines a vector YMAX(I) such that YMAX(I) is the maximum value (in modulus) that Y(I) has achieved in the past, i.e. for $t' < t$. A relative error control is used in this code, that is the quotient E(I)/YMAX(I) is compared with a specified tolerance factor. (The code uses an L_2 norm so that the actual situation is somewhat more involved.) Now consider the present problem as time approaches 6:00 p.m. The concentration c^1 has reached a peak value of about 10^8 at noon; thus YMAX(I) $\sim 10^8$ during the entire afternoon. Moreover as we approach 6:00 p.m., the concentrations c^1 fall to zero very rapidly; hence we are using the error criterion $10^{-8} \cdot E(I) < tol$ which is a very loose error control. In order to correct this situation we could put a floor value on the error control. For example we could define YMAX(I) as follows:

$$YMAX(I) = MAX\{|Y(I)|, 10^{-20}\}.$$

This would have the effect of replacing a relative error control by an absolute error control when $|Y(I)|$ is small. This change could have been made to

GEAR; however, we can also achieve this change through a user routine. Of course, this requires knowledge of the GEAR code and the DISPL code; so it is not a casual procedure.

To achieve the above change one must modify the definitions of YMAX(I) as described above. Moreover, this must be done after the GEAR program has defined YMAX. The GEAR program defines YMAX in subroutine DRIVE. The subroutine DRIVE calls a subroutine STIFF which in turn calls a DISPL subroutine GFUN. This latter subroutine calls several user routines such as VEL, DIFUSE, EXTSRC. Thus we can modify the definition of YMAX in any one of these subroutines, for example, subroutine VEL. To modify the definition of YMAX, we must have YMAX(I) and Y(I) available in this subroutine. The array YMAX(I) is in the common block GEAR2 and Y is in the common block GEAR10. These common blocks are invoked in subroutine VEL and the modification to YMAX(I) is made in this subroutine.

For this problem we use the following data in the namelists.

GRID

```
KR=1,KZ=4,
NSPEC=2,
ZLØW=30.0,ZUP=50.0,
CØNTZ=2,
NMZ=7,
ZMESH=32.5,35.0,37.5,40.0,42.5,45.0,47.5,
NQR=1,NQZ=4,
INITSW=T,TRANSW=T,GUESSW=F,STEDSW=F,
JZGRD=3,
ZGRID=30.,40.,50.,
```

DATA

```
NUTØUT=2,
NUFREQ=12,
UTØUT=0.0,86400.0,
EPS=1.D-3,HINIT=1.D-4,
GRAPH=T,
```

For the user routines, we used the following statements.

Subroutine RHØCP

```
RC=1.DO
```

Subroutine DIFUSE

```
      DIFUR=0.D0
      DIFUZ=(1.D-8)*DEXP(ZZ/5.D0)
```

Subroutine VEL

```
      CØMMØN/GEAR2/YMAX(1)
      CØMMØN/GEAR10/Y(1)
      DØUBLE PRECISIØN Y,YMAX
      DATA N/36/
      DØ 5 I=1,N
    5 YMAX(I)=DMAX1(DABS(Y(I)),1.D-20)
      VELR=0.D0
      VELZ=0.D0
```

The use of this subroutine to modify YMAX was just a matter of convenience.

Subroutine EXTSRC

In this routine we form the sources

$$R^1 = -k_1 c^1 c^3 - k_2 c^1 c^2 + 2k_3 c^3 + k_4 c^2$$

$$R^2 = k_1 c^1 c^3 - k_2 c^1 c^2 - k_4 c^2$$

Subroutine FDEXTU

In this subroutine we calculate the Frechet derivative of the distributed source.

For species 1 we have

$$UU(1) = -k_1 c^3 - k_2 c^2,$$

$$UU(2) = -k_2 c^1 + k_4.$$

For species 2 we have

$$UU(1) = k_1 c^3 - k_2 c^2$$

$$UU(2) = -k_2 c^1 - k_4$$

For both species, UUR(K) and UUZ(K) are zero for K=1,2.

Subroutine INDATA

In this routine we return the initial distributions

$$c^1(z,0) = 10^6 \gamma(z), \quad c^2(z,0) = 10^{12} \gamma(z) \quad \text{where}$$

$$\gamma(z) = 1 - \left(\frac{z-40}{10}\right)^2 + \frac{1}{2}\left(\frac{z-40}{10}\right)^4.$$

Subroutine BRHØ

Since $\frac{\partial c^i}{\partial z}(30,t) = \frac{\partial c^i}{\partial z}(50,t) = 0$, we just return RHØV=0.D0 on sides 2 and 4 for both species.

In this problem we used the z axis for the spatial variable and Hermite cubic (KZ=4,CØNTZ=2) B-splines. For spatially smooth solutions, the use of Hermite cubics is not an advantage; however, we use them here for illustrative purposes. The output is given at 30, 40, and 50 km at two-hour intervals during a 24-hour day. This problem was taken from Ref. [7] where a finite difference solution is given at the above spatial and time values. This data is given in Table 7.3.1.

The following pages contain the computational phase output for this problem.

TABLE 7.3.1. Finite Difference Solution for Atmospheric Model

Time	30 km	40 km	50 km
8:00 AM	$2.10 \cdot 10^{4}$ $5.06 \cdot 10^{11}$	$4.12 \cdot 10^{4}$ $9.96 \cdot 10^{11}$	$2.22 \cdot 10^{4}$ $5.36 \cdot 10^{11}$
10:00 AM	$1.31 \cdot 10^{7}$ $5.09 \cdot 10^{11}$	$2.54 \cdot 10^{7}$ $9.91 \cdot 10^{11}$	$1.44 \cdot 10^{7}$ $5.60 \cdot 10^{11}$
12:00	$4.74 \cdot 10^{7}$ $5.49 \cdot 10^{11}$	$8.68 \cdot 10^{7}$ $10.2 \cdot 10^{11}$	$5.29 \cdot 10^{7}$ $6.10 \cdot 10^{11}$
2:00 PM	$1.52 \cdot 10^{7}$ 5.91	$2.72 \cdot 10^{7}$ $10.6 \cdot 10^{11}$	$1.72 \cdot 10^{7}$ $6.71 \cdot 10^{11}$
4:00 PM	$2.46 \cdot 10^{4}$ $5.94 \cdot 10^{11}$	$4.37 \cdot 10^{4}$ $10.5 \cdot 10^{11}$	$2.84 \cdot 10^{4}$ $6.84 \cdot 10^{11}$
6:00 PM	0 $5.96 \cdot 10^{11}$	0 $10.5 \cdot 10^{11}$	0 $6.95 \cdot 10^{11}$
8:00 PM	0 $5.97 \cdot 10^{11}$	0 $10.5 \cdot 10^{11}$	0 $7.05 \cdot 10^{11}$
10:00 PM	0 $5.99 \cdot 10^{11}$	0 $10.4 \cdot 10^{11}$	0 $7.14 \cdot 10^{11}$
12:00	0 $6.01 \cdot 10^{11}$	0 $10.4 \cdot 10^{11}$	0 $7.22 \cdot 10^{11}$
2:00 AM	0 $6.03 \cdot 10^{11}$	0 $10.3 \cdot 10^{11}$	0 $7.29 \cdot 10^{11}$
4:00 AM	0 $6.04 \cdot 10^{11}$	0 $10.3 \cdot 10^{11}$	0 $7.36 \cdot 10^{11}$
6:00 AM	0 $6.06\ 10^{11}$	0 $10.2 \cdot 10^{11}$	0 $7.42 \cdot 10^{11}$

```
STORAGE MAXIMA FOR THIS COMPILATION :
MAXBRK          30
MAXSP            2
MAXTQD           4
MAXK             4
MXNRMZ         100
MXNVAR         100
MAXNOT          40
MXRGRD          20
MXZGRD          20
```

```
 READING NAMELIST  GRID


KR =  1    KZ =  4
NTIR =  0 NTIZ =  0
NSPEC =  2
RLOW =    0.0                                RUP =    0.1000000000000000D+01
ZLOW =    0.3000000000000000D+02             ZUP =    0.5000000000000000D+02
NMR =  0     NMZ =  7
 INITIAL CONTR = 13          INITIAL CONTZ =  2
GEOMETRY INDICATOR =  0
NO ADDITIONAL NON-INTERFACE R MESH POINTS
ADDITIONAL NON-INTERFACE Z MESH POINTS
I =  1 ZMESH(I) =    0.3250000000000000D+02
I =  2 ZMESH(I) =    0.3500000000000000D+02
I =  3 ZMESH(I) =    0.3750000000000000D+02
I =  4 ZMESH(I) =    0.4000000000000000D+02
I =  5 ZMESH(I) =    0.4250000000000000D+02
I =  6 ZMESH(I) =    0.4500000000000000D+02
I =  7 ZMESH(I) =    0.4750000000000000D+02
QUADRATURE ORDER FOR R DIRECTION     1
QUADRATURE ORDER FOR Z DIRECTION     4
MATERIAL TABLE IS GIVEN AS MATL(RINDEX,ZINDEX)
MATERIAL TABLE FOR ZINDEX =   1
 1
CONSRV=  F
ALGBCS=  T
LOGICAL SWITCHES TO CONTROL PROGRAM
STEDSW =  F      GUESSW =  F      TRANSW =  T      INITSW =  T
ISTDRS =  F          ITRARS =  F          IANAL =  F
DUMPSW =  F
IREVLA( 1) =  F
IREVLA( 2) =  F
ORDER OF SPLINE DERIVATIVES COMPUTED IS  0
NUMBER OF USER SUPPLIED POINTS IN R DIRECTION  1
RGRID( 1) =       0.50000000D+00
NUMBER OF USER SUPPLIED POINTS IN Z DIRECTION  3
ZGRID( 1) =       0.30000000D+02
ZGRID( 2) =       0.40000000D+02
ZGRID( 3) =       0.50000000D+02


 READING NAMELIST  DATA


CONTR TOO HIGH, BEING RESET TO      1-1

NVGAP =  0     NHGAP =  0     LR =  1     LZ =  8     NR =  1     NZ = 18

IL(I) =
 1

JL(J) =
 4   6   8  10  12  14  16  18

IREF(I) =
 1

JREF(J) =  1   1   1   1   1   1   1   1
MLTAB( 1, 1) =  1
```

```
MLTAB( 1, 2) =  1
MLTAB( 1, 3) =  1
MLTAB( 1, 4) =  1
MLTAB( 1, 5) =  1
MLTAB( 1, 6) =  1
MLTAB( 1, 7) =  1
MLTAB( 1, 8) =  1

HORIZONTAL ORDERING



NI=   2          NJ=   2          NCC=  -4          MBW=   7



NIH=   1          NJH=   1          NCCH=  -1          DM=   3


SYSTEM SIZE FOR THIS CASE
LR =     1     LZ =     8
NR =     1     NZ =    18
NVAR =   36

 THIS IS THE DIRECT VERSION

 THIS VERSION DOES NOT REQUIRE BOUNDARY CONDITIONS ON EVERY SIDE
FOR SPECIES NO.  1
 SIDE  1 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  2 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  3 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0
 SIDE  4 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0
FOR SPECIES NO.  2
 SIDE  1 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  2 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  3 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0
 SIDE  4 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  0     NS2=  0     NS3=  0     NS4=  0
 FOR SPECIES NO.  2     NS1=  0     NS2=  0     NS3=  0     NS4=  0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
SPECIES NO.  1 MATERIAL INDEX  1 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
SPECIES NO.  2 MATERIAL INDEX  1 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
SPECIES NO.  1 MATERIAL INDEX  1 HU2 =    0.10000000000000000D+01      HU4 =    0.10000000000000000D+01
SPECIES NO.  2 MATERIAL INDEX  1 HU2 =    0.10000000000000000D+01      HU4 =    0.10000000000000000D+01
REACTION RATES
FIRST ORDER RATES
CK INTO  1 FROM  1 IS    0.0
CK INTO  2 FROM  1 IS    0.0
CK INTO  1 FROM  2 IS    0.0
CK INTO  2 FROM  2 IS    0.0
SECOND ORDER REACTION RATES ARE
CKK INTO K =  1 FOR KP =  1 INTO KPP =  1          CKK( 1, 1, 1) =    0.0
CKK INTO K =  1 FOR KP =  2 INTO KPP =  1          CKK( 1, 2, 1) =    0.0
CKK INTO K =  1 FOR KP =  1 INTO KPP =  2          CKK( 2, 1, 1) =    0.0
CKK INTO K =  1 FOR KP =  2 INTO KPP =  2          CKK( 2, 2, 1) =    0.0
```

```
CKK INTO K =  2 FOR KP =            KPP =  1          CKK( 1, 1, 2) =    0.0
CKK INTO K =  2 FOR KP =  2 INTO KPP =  1          CKK( 1, 2, 2) =    0.0
CKK INTO K =  2 FOR KP =  1 INTO KPP =  2          CKK( 2, 1, 2) =    0.0
CKK INTO K =  2 FOR KP =  2 INTO KPP =  2          CKK( 2, 2, 2) =    0.0

ISTDFQ = 100
TIME AND SPACE GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  2
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL12
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES
   0.0                    0.8640000000000000D+05
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  F
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
EPS =     0.1000000000000000D-02      HINIT =     0.1000000000000000D-03      MP =  21     MXGORD =  5
CONTINUITY FOR R AND Z DIRECTIONS MAY HAVE BEEN RESET
CONTR =  0         CONTZ =  2
DEFAULT INITIAL COEFFICIENTS

        0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01
        0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01
        0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01
        0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01
        0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01
        0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01
        0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01
        0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01
        0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01      0.1000000000000000D+01

 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.5845507605104101D+06
   0.5470729890168923D+06
   0.5125313397923257D+06
   0.5005823263587777D+06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.5845507605104100D+12
   0.5470729890168923D+12
   0.5125313397923257D+12
   0.5005823263587777D+12
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  2)-TH RECTANGLE.
(FROM INIFIT)
   0.7682092330596028D+06
   0.7182242384883945D+06
   0.6536905503209353D+06
   0.6073443721740726D+06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  2)-TH RECTANGLE.
(FROM INIFIT)
   0.7682092330596027D+12
   0.7182242384883946D+12
   0.6536905503209353D+12
   0.6073443721740726D+12
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  3)-TH RECTANGLE.
(FROM INIFIT)
   0.9311287513666886D+06
   0.8955229152425516D+06
```

```
   0.8408606696138081D+06
   0.7942432856181262D+06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  3)-TH RECTANGLE.
(FROM INIFIT)
   0.9311287513666888D+12
   0.8955229152425518D+12
   0.8408606696138081D+12
   0.7942432856181262D+12
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  4)-TH RECTANGLE.
(FROM INIFIT)
   0.9997548357817510D+06
   0.9931863542315364D+06
   0.9723089106323048D+06
   0.9473985950662583D+06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  4)-TH RECTANGLE.
(FROM INIFIT)
   0.9997548357817508D+12
   0.9931863542315363D+12
   0.9723089106323048D+12
   0.9473985950662584D+12
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  5)-TH RECTANGLE.
(FROM INIFIT)
   0.9473985950662591D+06
   0.9723089106323055D+06
   0.9931863542315357D+06
   0.9997548357817505D+06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  5)-TH RECTANGLE.
(FROM INIFIT)
   0.9473985950662596D+12
   0.9723089106323060D+12
   0.9931863542315357D+12
   0.9997548357817502D+12
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  6)-TH RECTANGLE.
(FROM INIFIT)
   0.7942432856181271D+06
   0.8408606696138088D+06
   0.8955229152425512D+06
   0.9311287513666884D+06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  6)-TH RECTANGLE.
(FROM INIFIT)
   0.7942432856181271D+12
   0.8408606696138086D+12
   0.8955229152425512D+12
   0.9311287513666887D+12
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  7)-TH RECTANGLE.
(FROM INIFIT)
   0.6073443721740731D+06
   0.6536905503209357D+06
   0.7182242384883944D+06
   0.7682092330596029D+06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  7)-TH RECTANGLE.
(FROM INIFIT)
   0.6073443721740730D+12
   0.6536905503209356D+12
   0.7182242384883945D+12
   0.7682092330596030D+12
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  8)-TH RECTANGLE.
(FROM INIFIT)
   0.5005823263587777D+06
   0.5125313397923259D+06
   0.5470729890168923D+06
   0.5845507605104101D+06
```

```
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  8)-TH RECTANGLE.
(FROM INIFIT)
   0.5005823263587776D+12
   0.5125313397923258D+12
   0.5470729890168923D+12
   0.5845507605104102D+12
```

```
READING NAMELIST  DATA

 CHANGES IN NAMELIST DA      VE BEEN MADE FOR TRANSIENT
FOR SPECIES NO.  1
 SIDE  1 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  2 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  3 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0
 SIDE  4 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0
FOR SPECIES NO.  2
 SIDE  1 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  2 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  3 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0
 SIDE  4 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  0     NS2=  0     NS3=  0     NS4=  0
 FOR SPECIES NO.  2     NS1=  0     NS2=  0     NS3=  0     NS4=  0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
SPECIES NO.  1 MATERIAL INDEX  1 HU1 =    0.1000000000000000D+01      HU3 =    0.1000000000000000D+01
SPECIES NO.  2 MATERIAL INDEX  1 HU1 =    0.1000000000000000D+01      HU3 =    0.1000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
SPECIES NO.  1 MATERIAL INDEX  1 HU2 =    0.1000000000000000D+01      HU4 =    0.1000000000000000D+01
SPECIES NO.  2 MATERIAL INDEX  1 HU2 =    0.1000000000000000D+01      HU4 =    0.1000000000000000D+01
REACTION RATES
FIRST ORDER RATES
CK INTO  1 FROM  1 IS    0.0
CK INTO  2 FROM  1 IS    0.0
CK INTO  1 FROM  2 IS    0.0
CK INTO  2 FROM  2 IS    0.0
SECOND ORDER REACTION RATES ARE
CKK INTO K =  1 FOR KP =  1 INTO KPP =  1        CKK( 1, 1, 1) =    0.0
CKK INTO K =  1 FOR KP =  2 INTO KPP =  1        CKK( 1, 2, 1) =    0.0
CKK INTO K =  1 FOR KP =  1 INTO KPP =  2        CKK( 2, 1, 1) =    0.0
CKK INTO K =  1 FOR KP =  2 INTO KPP =  2        CKK( 2, 2, 1) =    0.0
CKK INTO K =  2 FOR KP =  1 INTO KPP =  1        CKK( 1, 1, 2) =    0.0
CKK INTO K =  2 FOR KP =  2 INTO KPP =  1        CKK( 1, 2, 2) =    0.0
CKK INTO K =  2 FOR KP =  1 INTO KPP =  2        CKK( 2, 1, 2) =    0.0
CKK INTO K =  2 FOR KP =  2 INTO KPP =  2        CKK( 2, 2, 2) =    0.0

ISTDFQ = 100
TIME AND SPACE GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  2
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL12
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES
   0.0                    0.8640000000000000D+05
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  F
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
EPS =    0.1000000000000000D-02      HINIT =    0.1000000000000000D-03      MF =  21      MXGORD =  5
CONTINUITY FOR R AND Z DIRECTIONS
CONTR =  0          CONTZ =  2
INITIAL COEFFICIENTS FOR TRANSIENT

       0.4999264413423072D+06       0.4999264413423071D+12       0.5003222170878941D+06       0.5003222170878942D+12
       0.5409738759634485D+06       0.5409738759634484D+12       0.6505195792121346D+06       0.6505195792121346D+12
       0.7187855915021812D+06       0.7187855915021813D+12       0.8438328868260228D+06       0.8438328868260226D+12
       0.9004477533441054D+06       0.9004477533441058D+12       0.9785851462945553D+06       0.9785851462945552D+12
       0.1000064170975785D+07       0.1000064170975785D+13       0.1000064170975782D+07       0.1000064170975782D+13
```

0.9785851462945581D+06	0.9785851462945591D+12	0.9004477533441027D+06	0.9004477533441026D+12
0.8438328868260250D+06	0.8438328868260246D+12	0.7187855915021796D+06	0.7187855915021800D+12
0.6505195792121356D+06	0.6505195792121353D+12	0.5409738759634476D+06	0.5409738759634480D+12
0.5003222170878950D+06	0.5003222170878946D+12	0.4999264413423069D+06	0.4999264413423070D+12

```
 BEGIN TRANSIENT SOLUTION


PROUT FOR TIME =        0.0

  W =
      0.49992644D+06      0.49992644D+12      0.50032222D+06      0.50032222D+12      0.54097388D+06      0.54097388D+12
      0.65051958D+06      0.65051958D+12      0.71878559D+06      0.71878559D+12      0.84383289D+06      0.84383289D+12
      0.90044775D+06      0.90044775D+12      0.97858515D+06      0.97858515D+12      0.10000642D+07      0.10000642D+13
      0.10000642D+07      0.10000642D+13      0.97858515D+06      0.97858515D+12      0.90044775D+06      0.90044775D+12
      0.84383289D+06      0.84383289D+12      0.71878559D+06      0.71878559D+12      0.65051958D+06      0.65051958D+12
      0.54097388D+06      0.54097388D+12      0.50032222D+06      0.50032222D+12      0.49992644D+06      0.49992644D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.4999264413423069D+06

APPROXIMATE SOLUTION
   0.1000064170975784D+07

APPROXIMATE SOLUTION
   0.4999264413423072D+06
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.4999264413423069D+12

APPPOXIMATE SOLUTION
   0.1000064170975784D+13

APPROXIMATE SOLUTION
   0.4999264413423071D+12
   TOUT=   0.0


PROUT FOR TIME =        0.72000000D+04

  W =
      0.20761057D+05      0.50111956D+12      0.20778145D+05      0.50153205D+12      0.22467506D+05      0.54231046D+12
      0.27001837D+05      0.65176358D+12      0.29824085D+05      0.71989004D+12      0.34977656D+05      0.84429451D+12
      0.37299108D+05      0.90033404D+12      0.40468972D+05      0.97685474D+12      0.41308091D+05      0.99711135D+12
      0.41192268D+05      0.99431537D+12      0.40226237D+05      0.97099507D+12      0.36866378D+05      0.88988798D+12
      0.34466661D+05      0.83195927D+12      0.29365839D+05      0.70892837D+12      0.26722364D+05      0.64501745D+12
      0.23043523D+05      0.55621476D+12      0.22175029D+05      0.53525069D+12      0.22201937D+05      0.53590019D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.2220193664096921D+05

APPROXIMATE SOLUTION
   0.4125017945629122D+05

APPROXIMATE SOLUTION
   0.2076105699898301D+05
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
```

```
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.53590C1853939819D+12

APPROXIMATE SOLUTION
   0.9957133627759844D+12

APPROXIMATE SOLUTION
   0.5011195625014761D+12
   TOUT=   0.7200000000000000D+04  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.22790000D+04
TIME LEFT       0.17535000D+05


PROUT FOR TIME =       0.14400000D+05

  W =
      0.12908690D+08      0.50256248D+12      0.12919537D+08      0.50298663D+12      0.13965495D+08      0.54388511D+12
      0.16761929D+08      0.65323157D+12      0.18500455D+08      0.72121258D+12      0.21664403D+08      0.84493331D+12
      0.23082767D+08      0.90039686D+12      0.24996354D+08      0.97522620D+12      0.25483258D+08      0.99426639D+12
      0.25338443D+08      0.98860347D+12      0.24696308D+08      0.96349304D+12      0.22554066D+08      0.87972254D+12
      0.21052400D+08      0.82100180D+12      0.18020727D+08      0.70245396D+12      0.16574036D+08      0.64588471D+12
      0.14733358D+08      0.57391014D+12      0.14353287D+08      0.55904869D+12      0.14362514D+08      0.55940946D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.1436251371761089D+08

APPROXIMATE SOLUTION
   0.2541085034277794D+08

APPROXIMATE SOLUTION
   0.1290869C042715 0D+08
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.5594094582439390D+12

APPROXIMATE SOLUTION
   0.9914349285040650D+12

APPROXIMATE SOLUTION
   0.5025624774906308D+12
   TOUT=   0.1440000000000000D+05  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.65000000D+02
TIME LEFT       0.17470000D+05


PROUT FOR TIME =       0.21600000D+05

  W =
      0.46829963D+08      0.54269462D+12      0.46865844D+08      0.54312753D+12      0.50263702D+08      0.58412346D+12
      0.59130440D+08      0.69330767D+12      0.64911875D+08      0.76110231D+12      0.75120973D+08      0.88404200D+12
      0.79666138D+08      0.93898406D+12      0.85719261D+08      0.10119220D+13      0.87193653D+08      0.10297127D+13
```

```
      0.86485436D+08      0.10211681D+13      0.84259683D+08      0.99431225D+12      0.77144314D+08      0.90845849D+12
      0.72285038D+08      0.84982666D+12      0.63019632D+08      0.73803042D+12      0.58909990D+08      0.68944363D+12
      0.53891859D+08      0.62789528D+12      0.52922527D+08      0.61619946D+12      0.52941712D+08      0.61643094D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.5294171211892320D+08

APPROXIMATE SOLUTION
   0.8683954413069259D+08

APPROXIMATE SOLUTION
   0.4682996301832602D+08
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.6164309401435280D+12

APPROXIMATE SOLUTION
   0.1025440392760540D+13

APPROXIMATE SOLUTION
   0.5426946248644204D+12
   TOUT=   0.2160000000000000D+05  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.29000000D+02
TIME LEFT       0.17441000D+05


PROUT FOR TIME =       0.28800000D+05

  W =
      0.14969979D+08      0.58302301D+12      0.14981232D+08      0.58346290D+12      0.16032302D+08      0.62455270D+12
      0.18821159D+08      0.73357993D+12      0.20550515D+08      0.80118807D+12      0.23674942D+08      0.92333751D+12
      0.25061639D+08      0.97755126D+12      0.26883157D+08      0.10487649D+13      0.27305676D+08      0.10652837D+13
      0.27013632D+08      0.10538653D+13      0.26283501D+08      0.10253197D+13      0.24052492D+08      0.93809646D+12
      0.22579447D+08      0.88050751D+12      0.19908479D+08      0.77608782D+12      0.18788753D+08      0.73231391D+12
      0.17451115D+08      0.68002149D+12      0.17203329D+08      0.67033493D+12      0.17207415D+08      0.67049463D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.1720741494294180D+08

APPROXIMATE SOLUTION
   0.2715965397759396D+08

APPROXIMATE SOLUTION
   0.1496997930484354D+08
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.6704946343001749D+12

APPROXIMATE SOLUTION
   0.1059574509224302D+13
```

```
APPROXIMATE SOLUTION
   0.5830230691920294D+12
   TOUT=   0.2880000000000000D+05  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.17800000D+03
TIME LEFT       0.17263000D+05


PROUT FOR TIME =       0.36000000D+05

  W =
      0.24231271D+05      0.58449254D+12      0.24249749D+05      0.58493829D+12      0.25957381D+05      0.62613056D+12
      0.30472392D+05      0.73504551D+12      0.33268451D+05      0.80249533D+12      0.38301332D+05      0.92390628D+12
      0.40523477D+05      0.97751316D+12      0.43400314D+05      0.10469143D+13      0.44C317C2D+05      0.10621461D+13
      0.43440430D+05      0.10478820D+13      0.42190936D+05      0.10177389D+13      0.38549831D+05      0.92990094D+12
      0.36232941D+05      0.87409000D+12      0.32201044D+05      0.77674611D+12      0.30583625D+05      0.73772907D+12
      0.28676764D+05      0.69173001D+12      0.28333922D+05      0.68345973D+12      0.28338678D+05      0.68357446D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.2833867787659093D+05

APPROXIMATE SOLUTION
   0.4373606636063932D+05

APPROXIMATE SOLUTION
   0.2423127065620274D+05
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.6835744560403455D+12

APPROXIMATE SOLUTION
   0.1055014053688299D+13

APPROXIMATE SOLUTION
   0.5844925418915659D+12
   TOUT=   0.3600000000000000D+05  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.20300000D+03
TIME LEFT       0.17060000D+05


PROUT FOR TIME =       0.43200000D+05

  W =
     -0.21014397D-32      0.58570450D+12     -0.21030572D-32      0.58615507D+12     -0.22512015D-32      0.62744470D+12
     -0.26417500D-32      0.73624719D+12     -0.28833606D-32      0.80353557D+12     -0.3317C183D-32      0.92419502D+12
     -0.35077433D-32      0.97718292D+12     -0.37518285D-32      0.10447384D+13     -0.38029635D-32      0.10586571D+13
     -0.37426048D-32      0.10416156D+13     -0.36278360D-32      0.10100597D+13     -C.33C74205D-32      0.92237716D+12
     -0.31120793D-32      0.86858686D+12     -0.27878237D-32      0.77800545D+12     -0.26639527D-32      0.74291685D+12
     -0.25194590D-32      0.70189792D+12     -0.24942474D-32      0.69468753D+12     -0.24945257D-32      0.69477214D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )
```

```
APPROXIMATE SOLUTION
  -0.2494525718306979D-32

APPROXIMATE SOLUTION
  -0.3772784142334187D-32

APPROXIMATE SOLUTION
  -0.2101439658958643D-32
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.6947721404634738D+12

APPROXIMATE SOLUTION
   0.1050136336078147D+13

APPROXIMATE SOLUTION
   0.5857044962281630D+12
   TOUT=   0.4320000000000000D+05  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.14290000D+04
TIME LEFT       0.15631000D+05


PROUT FOR TIME =       0.50400000D+05

  W =
      0.35830007D-33      0.58691929D+12      0.35857567D-33      0.58737383D+12      0.38383273D-33      0.62875501D+12
      0.45041975D-33      0.73744049D+12      0.49161185D-33      0.80456135D+12      0.56553918D-33      0.92444982D+12
      0.59804735D-33      0.97680245D+12      0.63963794D-33      0.10424769D+13      0.64833852D-33      0.10550586D+13
      0.63810459D-33      0.10353538D+13      0.61867277D-33      0.10026250D+13      0.56436148D-33      0.91572796D+12
      0.53111248D-33      0.86422227D+12      0.47554004D-33      0.77976944D+12      0.45415934D-33      0.74798418D+12
      0.42918450D-33      0.71104532D+12      0.42480838D-33      0.70466939D+12      0.42485843D-33      0.70473255D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PPOUT VIA TIMEX )

APPPOXIMATE SOLUTION
   0.4248584341367259D-33

APPROXIMATE SOLUTION
   0.6432215564252698D-33

APPROXIMATE SOLUTION
   0.3583000734683872D-33
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.7047325472749630D+12

APPROXIMATE SOLUTION
   0.1045206174614654D+13

APPROXIMATE SOLUTION
   0.5869192878265545D+12
   TOUT=   0.5040000000000000D+05  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.13000000D+02
TIME LEFT       0.15618000D+05
```

```
PROUT FOR TIME =        0.57600000D+05

  W =
     -0.33601807D-32       0.58813672D+12      -0.33627652D-32       0.58859452D+12      -0.35996303D-32       0.63006164D+12
     -0.42240895D-32       0.73862508D+12      -0.46103912D-32       0.80557224D+12      -0.53036736D-32       0.92466995D+12
     -0.56085238D-32       0.97637015D+12      -0.59985176D-32       0.10401304D+13      -0.60800783D-32       0.10513566D+13
     -0.59840431D-32       0.10291416D+13      -0.58017916D-32       0.99549984D+12      -0.52924975D-32       0.90989185D+12
     -0.49807432D-32       0.86073430D+12      -0.44596898D-32       0.78185854D+12      -0.42592249D-32       0.75287283D+12
     -0.40250599D-32       0.71932546D+12      -0.39840280D-32       0.71362016D+12      -0.39844973D-32       0.71366776D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
  -0.3984497290889737D-32

APPROXIMATE SOLUTION
  -0.6032060695688750D-32

APPROXIMATE SOLUTION
  -0.3360180682110748D-32
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.7136677567228190D+12

APPROXIMATE SOLUTION
   0.1040249081335329D+13

APPROXIMATE SOLUTION
   0.5881367214725579D+12
   TOUT=   0.5760000000000000D+05  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.60000000D+01
TIME LEFT       0.15612000D+05


PROUT FOR TIME =        0.64800000D+05

  W =
      0.3038613AD-27       0.58935656D+12      -0.71970727D-27       0.58981699D+12       0.25711574D-27       0.63136474D+12
     -0.75341548D-28       0.73980065D+12      -0.84226325D-28       0.80656783D+12       0.67842933D-28       0.92485467D+12
     -0.34948267D-29       0.97588470D+12       0.21425130D-28       0.10377023D+13      -0.10081947D-28       0.10475656D+13
      0.60858107D-29       0.10230224D+13      -0.28767135D-29       0.98871650D+12       0.17126473D-29       0.90477923D+12
     -0.83396883D-30       0.85796519D+12       0.48225267D-30       0.78415812D+12      -0.26736526D-30       0.75756311D+12
      0.15853991D-30       0.72686478D+12      -0.17207177D-30       0.72170530D+12       0.28781714D-31       0.72174159D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.2878171400293841D-31

APPROXIMATE SOLUTION
  -0.1998068324656972D-29

APPROXIMATE SOLUTION
```

```
   0.3038613830832379D-27
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.7217415928622239D+12

APPROXIMATE SOLUTION
   0.1035294006580380D+13

APPROXIMATE SOLUTION
   0.5893565638845197D+12
   TOUT=   0.6480000000000000D+05  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.15000000D+02
TIME LEFT       0.15597000D+05


PROUT FOR TIME =       0.72000000D+05

  W =
     0.12435132D-26      0.59057865D+12     -0.29452285D-26      0.59104114D+12      0.10522181D-26      0.63266437D+12
    -0.30828978D-27      0.74096695D+12     -0.34464615D-27      0.80754782D+12      0.27767C54D-27      0.92500340D+12
    -0.14297890D-27      0.97534519D+12      0.87720251D-28      0.10351956D+13     -0.41215333D-28      0.10436981D+13
     0.24947174D-28      0.10170247D+13     -0.11731379D-28      0.98228892D+12      0.70460292D-29      0.90032421D+12
    -0.33776693D-29      0.85581402D+12      0.20050135D-29      0.78660388D+12     -0.10640576D-29      0.76204910D+12
     0.67721692D-30      0.73374306D+12     -0.67603236D-30      0.72902665D+12      0.14592204D-30      0.72905501D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.1459220425798624D-30

APPROXIMATE SOLUTION
  -0.8134079675338683D-29

APPROXIMATE SOLUTION
   0.1243513217593859D-26
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.7290550109117528D+12

APPROXIMATE SOLUTION
   0.1030361394204485D+13

APPROXIMATE SOLUTION
   0.5905786519410204D+12
   TOUT=   0.7200000000000000D+05  DELTA T =   0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE       0.10000000D+01
TIME LEFT       0.15596000D+05


PROUT FOR TIME =       0.79200000D+05

  W =
     0.28284141D-26      0.59180276D+12     -0.66988365D-26      0.59226694D+12      0.23933193D-26      0.63396069D+12
```

```
      -0.70113172D-27      [illegible]D+12       -0.78381750D-27      0.80851184D+12      0.63164705D-27      [illegible]511549D+12
      -0.32511078D-27      [illegible]D+12        0.19961859D-27      0.10326123D+13     -0.93642735D-28      0.10397651D+13
       0.56841621D-28      0.10111937D+13        -0.26586406D-28      0.97625903D+12      0.16114147D-28      0.89644236D+12
      -0.75996351D-29      0.85411740D+12         0.46345561D-29      0.78906927D+12     -0.23493493D-29      0.76630449D+12
       0.16072673D-29      0.74009108D+12        -0.14713669D-29      0.73575751D+12      0.39816783D-30      0.73577950D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.3981678312493733D-30

APPROXIMATE SOLUTION
  -0.1840055695135602D-28

APPROXIMATE SOLUTION
   0.2828414105480834D-26
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.7357795044692915D+12

APPROXIMATE SOLUTION
   0.1025479379376244D+13

APPROXIMATE SOLUTION
   0.5918027643486777D+12
   TOUT=    0.7920000000000000D+05   DELTA T =    0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE        0.10000000D+01
TIME LEFT       0.15595000D+05


PROUT FOR TIME =       0.86400000D+05

  U =
       0.36058633D-27      0.59302869D+12        -0.85400307D-27      0.59349396D+12      0.30511817D-27      0.63525382D+12
      -0.89379795D-28      0.74327055D+12        -0.99920642D-28      0.80945949D+12      0.80531771D-28      0.92519017D+12
      -0.41440960D-28      0.97409880D+12         0.25455017D-28      0.10299567D+13     -0.11931556D-28      0.10357839D+13
       0.72529621D-29      0.10055667D+13        -0.33831253D-29      0.97064057D+12      0.20600397D-29      0.89304437D+12
      -0.96346771D-30      0.85274712D+12         0.59565550D-30      0.79147867D+12     -0.29490980D-30      0.77032616D+12
       0.20924981D-30      0.74600926D+12        -0.18327663D-30      0.74202432D+12      0.55062983D-31      0.74204049D+12

 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.5506298275070659D-31

APPROXIMATE SOLUTION
  -0.2339297002746397D-29

APPROXIMATE SOLUTION
   0.3605863339370921D-27
 VALUES OF CONCENTRATIONS U OF THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.7420404882437140D+12
```

```
APPROXIMATE SOLUTION
   0.1020675306413355D+13

APPROXIMATE SOLUTION
   0.5930286868305982D+12
   TOUT=    0.8640000000000000D+05  DELTA T =    0.8640000000000000D+05

ESTIMATED TIME FOR A CALL TO DRIVE        0.35000000D+02
TIME LEFT        0.15560000D+05

****** NORMAL DUMP AT END OF TIMEX
         I =      2          TOUT =     0.8640000000000000D+05


END OF CASE
```

READING NAMELIST GRID

To illustrate the use of the graphics programs for ITIME=T, we will present plots using CSP, CØNTØR, and THREED. In the case of CSP, we present a graph of each species on the same frame. At r = 30km, each species $c^i(z,t)$ is plotted as a function of t where t is evaluated at 2-hour intervals starting at 6:00 AM for one 24-hour period. Note that the data points on the graph are not connected with any type of curve. This is a limitation of the present version of CSP. In the case of CØNTØR, we present two contour plots -- one for each species. The horizontal axis is time ($0 \leq t \leq 86400$ sec) and the vertical axis is the spatial variable z ($30 \leq z \leq 50$). Note that the contour heights for the second species are too large for the printing capability of BLACKBØX. Hence we have ** printed for the contour heights. In the case of THREED we present two graphs -- one for each species. In each case the independent variables are t and z (as in CØNTØR), and the dependent variable is the species concentrations $c^i(z,t)$.

In the following we give the namelist data used for each of the three graphics programs.

For CSP:

Namelist FØRMAT

1. Indicator for iterative or direct version.
   ITRTV=0,
2. Number of curves produced
   We want a time plot for each of two species.
   IGNUM=2,
3. Logical indicator for time option.
   ITIME=T,
4. Number of time values.
   There are 13 output time values.
   NTIME=13,

Namelist CSPIN

1. Indicator for cinema mode.
   ICN=0, (Default)
2. Indicator for grouping format.
   We will put each curve on a separate axis; thus we want separate format.
   IFØRMT=0, (Default)

3. Species number for each curve.
   We will plot the first species on the first curve and the second species on the second curve.
   ISPEC=1,2,

4. Ordering of curves on frames.
   LØRDER=1,2,

5. Frame number indicator.
   Both curves will be plotted on the first frame.
   LGRØUP=1,1,

6. Estimate of minimum value of the ordinates.
   YAXMIN=0,0, (Default)

7. Estimate of maximum value of the ordinates.
   YAXMAX=0,0, (Default)

8. Coordinates of the point at which the solution is evaluated.
   A1=0.0,A2=0.0,
   B1=30.0,B2=30.0,

For CØNTØR:

Namelist FØRMAT is the same as in CSP. (We allow NRESIN to take on its default value of the Macro variable NRESD=21.)

Namelist CNTRIN

1. Species number for each frame.
   ISPEC=1,2,

2. Coordinates of line to be used as the spatial variable.
   RMIN=0.0,0.0, (Default)
   RMAX=0.0,0.0,
   ZMIN=30.0,30.0,
   ZMAX=50.0,50.0,

For THREED:

Namelist FØRMAT is the same as in CSP. (We again allow NRESIN=NRESD, its default value.)

Namelist DIM3IN

This namelist contains all the variables which appear in CNTRIN and these values are unchanged. In addition, we need the following

variables.

1. Coordinates of the viewpoint.

   RVIEW=-100.0,
   ZVIEW=-100.0,
   FVIEW=150.0,

2. Bounds on the function axis.

   FMATMN=0.0, FMATMX=1.0, (Defaults)

The graphs are presented on the following pages.

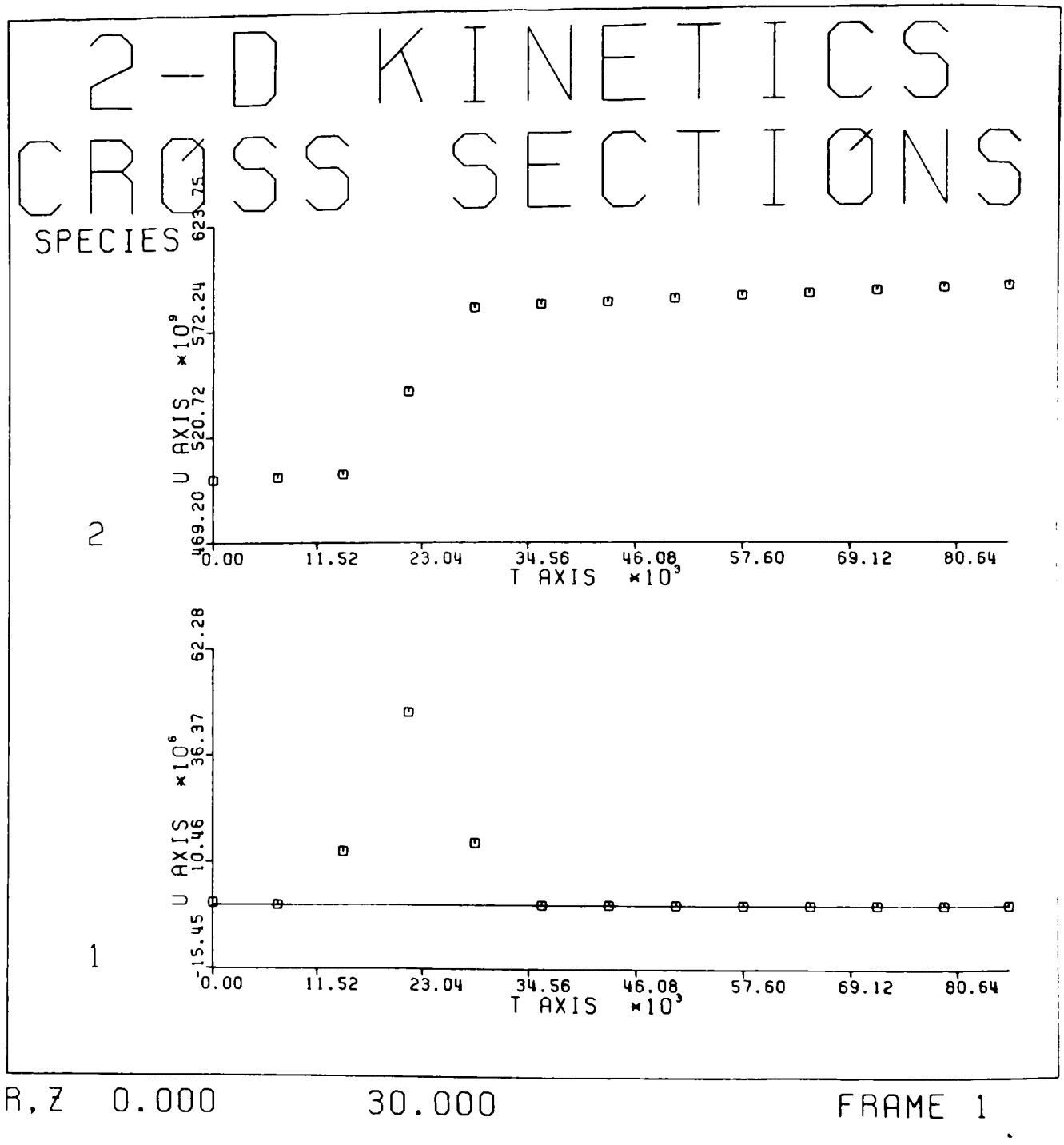


Figure 7.3.1
Time graphs of the species concentrations via CSP

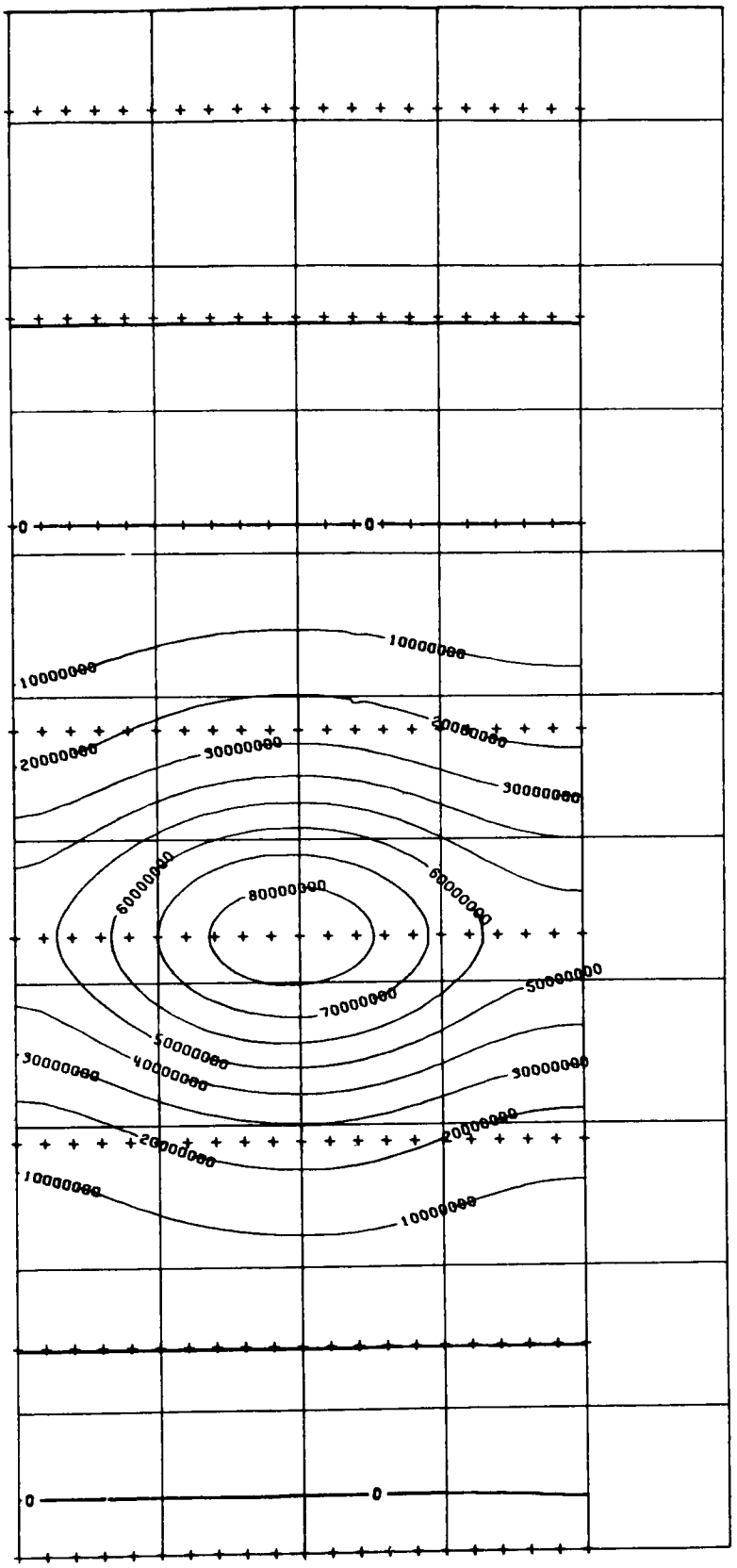


Figure 7.3.2

Time graph of the first species via CØNTØR

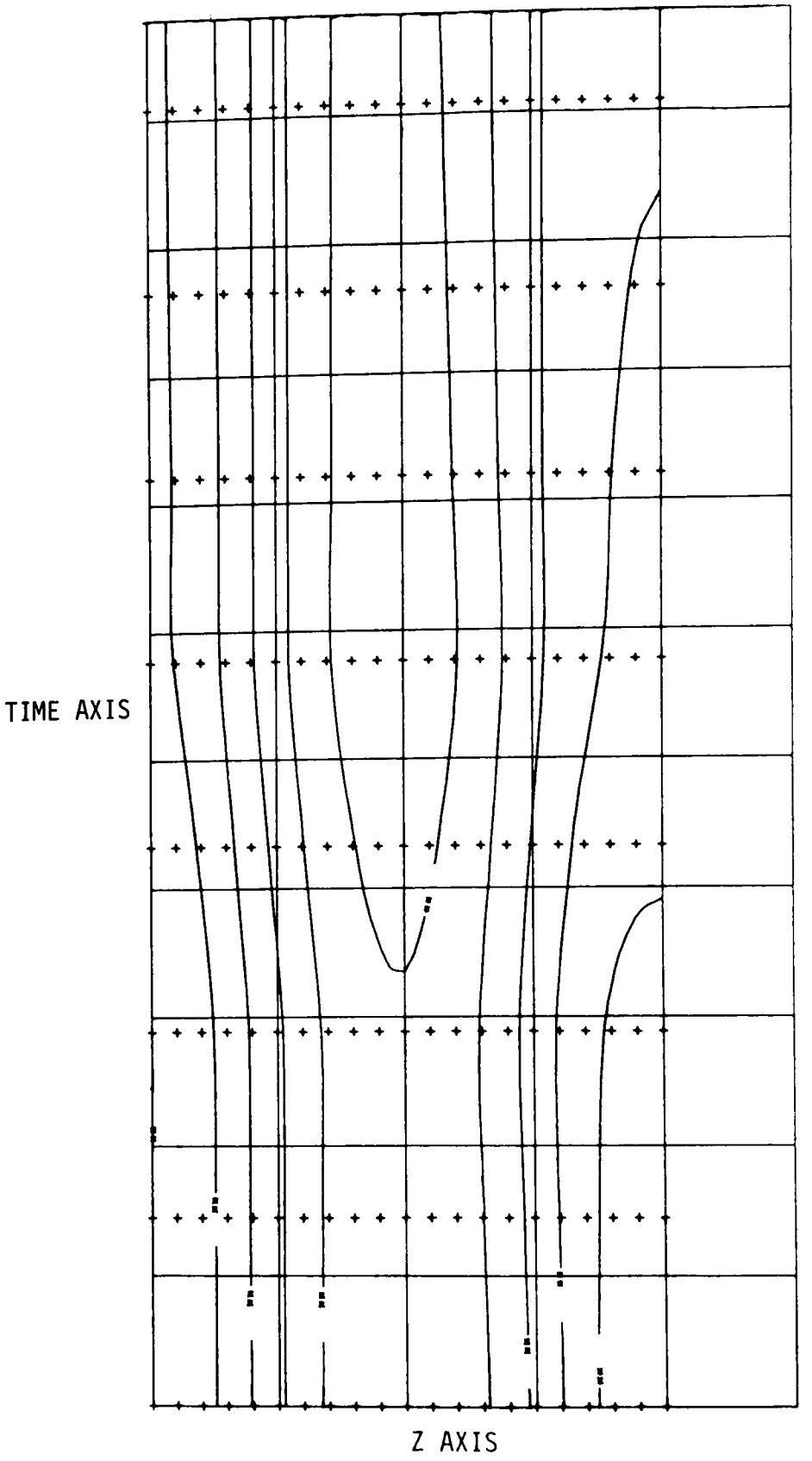
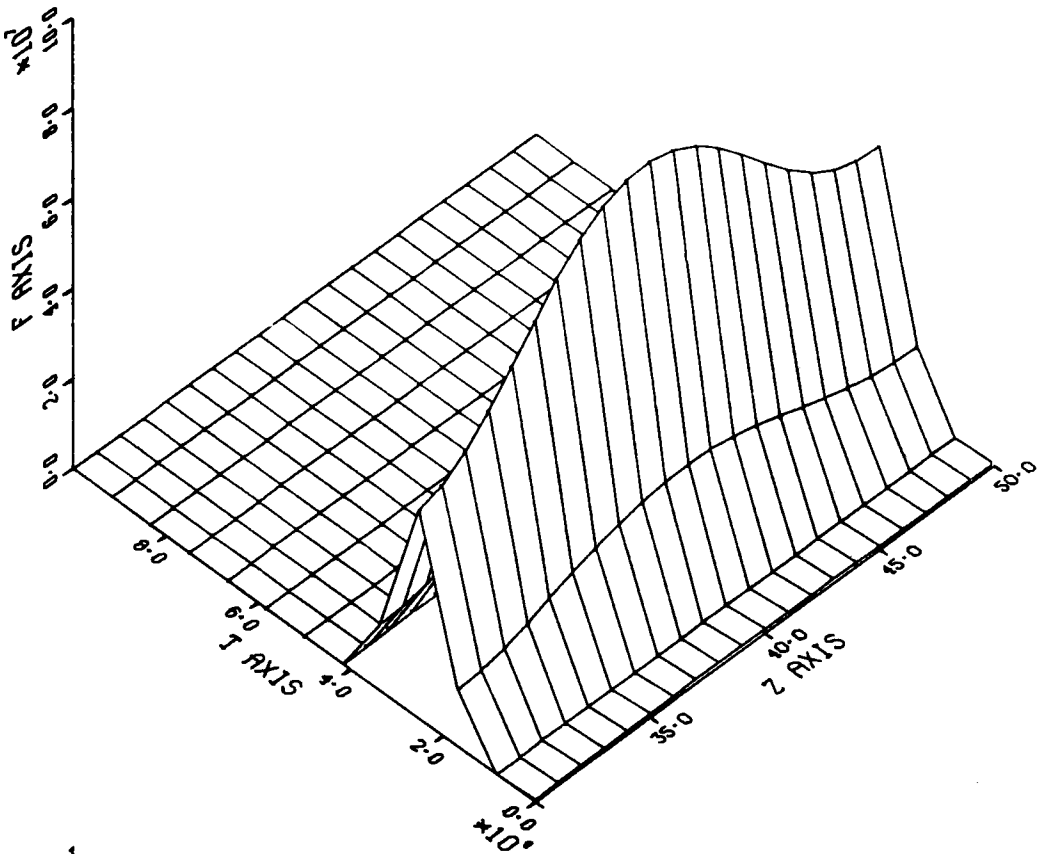


Figure 7.3.3
Time graph of the second species via CØNTØR

2-D KINETICS



FRAME 1
SPECIES 1

Figure 7.3.4
Time graph of the first species via THREED

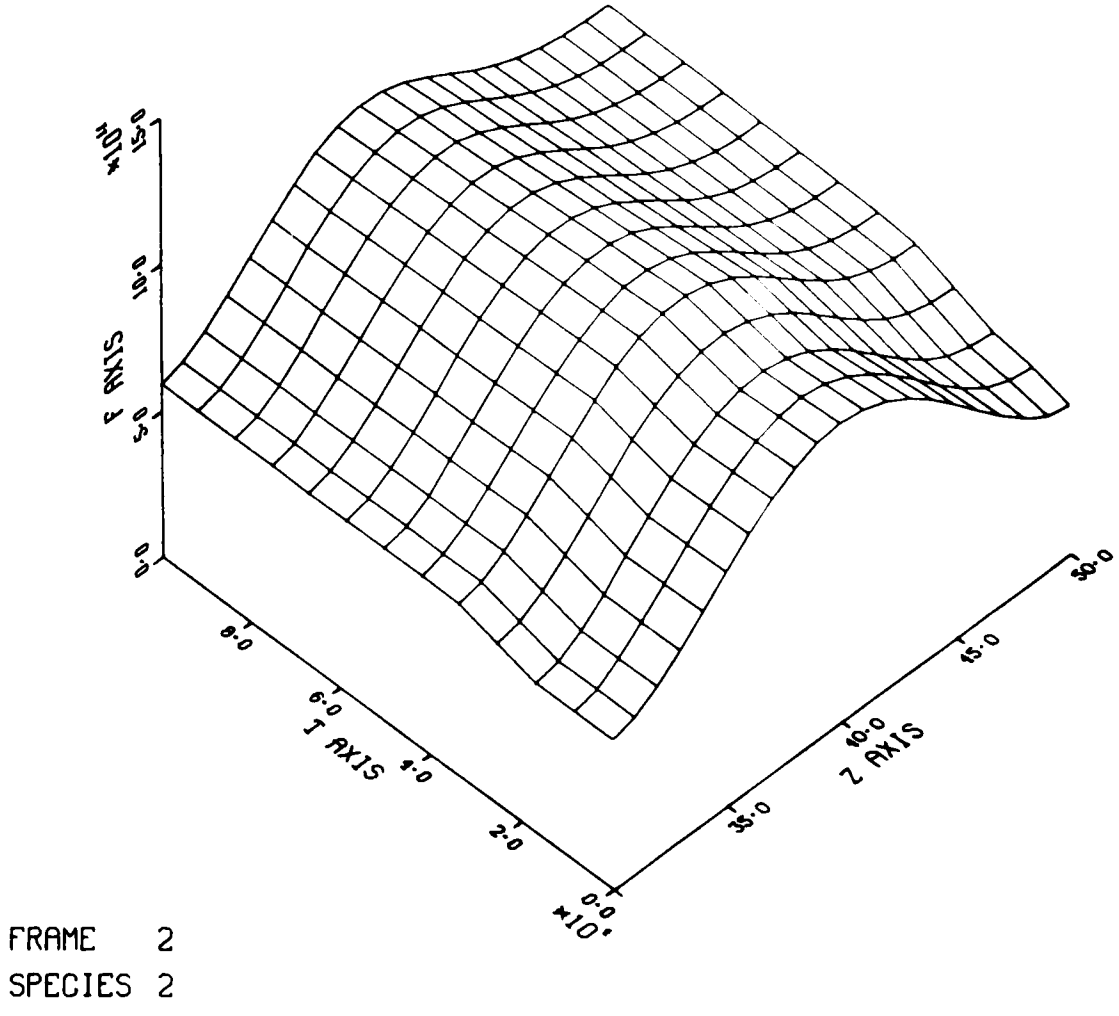


Figure 7.3.5
Time graph of the second species via THREED

7.4 Cooling of a Sphere in a Well Stirred Fluid

This problem will illustrate the use of spherical geometry and the differential boundary condition version of the code.

Consider a homogeneous sphere initially at a uniform temperature T, which is immersed in a volume V_f of a well stirred fluid at a temperature T_0 in an insulated tank. Let

k_s.....thermal conductivity of the sphere,

$\rho_s C_{ps}$.....heat capacity of the sphere,

$T_s = T_s(r,t)$.temperature of the sphere,

$T_f = T_f(t)$...temperature of the fluid at the surface of the sphere,

R.....radius of the sphere,

V_s.....volume of the sphere.

Define the following dimensionless variables.

$$\alpha_s = k_s/\rho_s C_{ps},$$

$$\xi = r/R,$$

$$\tau = \alpha_s t/R^2,$$

$$\theta_s = \theta_s(\xi,\tau) = (T_1-T_s)/(T_1-T_0),$$

$$\theta_f = (T_1-T_f)/(T_1-T_0).$$

For this problem, we will set $T_0 = 0$, then we have the following conduction problem.

(7.4.1) $$\frac{\partial\theta_s}{\partial\tau} = \frac{1}{\xi^2}\frac{\partial}{\partial\xi}\ \xi^2\ \frac{\partial\theta_s}{\partial\xi}\ ,\quad 0 \le \xi < 1$$

(7.4.2) $$\frac{\partial\theta_s}{\partial\xi}(0,\tau) = 0\ ,$$

(7.4.3) $$\theta_s(1,\tau) = \theta_f(\tau) \quad \text{where} \quad \frac{d\theta_f}{d\tau} = -\frac{3}{B}\frac{\partial\theta_s}{\partial\xi}(1,\tau),$$

$$B = \rho_f C_{pf} V_f/\rho_x C_{ps} V_s,$$

(7.4.4) $$\theta_s(\xi,0) \equiv 0 \quad \text{for} \quad 0 \le \xi < 1, \text{ and}$$

$$\theta_f(0) = 1.$$

Note that the boundary condition at $\xi = 1$ can be written in the form

$$\frac{\partial \theta_s}{\partial \tau}(1,\tau) = -\frac{3}{B}\frac{\partial \theta_s}{\partial \xi}(1,\tau) \quad . \tag{7.4.5}$$

This boundary condition is given in a time differential form; hence it is natural to use the differential boundary condition version to solve this problem.

This problem is selected from the text by Bird, Stewart, and Lightfoot, Ref. [8], page 357; the same problem can also be found in Carslaw and Jaeger, Ref. [5], page 205. The solution to this problem can be written in the following form (with $T_0 = 0$).

$$\theta_s(\xi,\tau) = \frac{B}{B+1} + \frac{2B}{3\xi}\sum_{k=1}^{\infty} \exp(-\tau b_k^2)\,\frac{[B^2 b_k^4 + 3(2B+3)b_k^2 + 9]}{B^2 b_k^4 + 9(B+1)b_k^2}\,\sin \xi b_k \sin b_k \tag{7.4.6}$$

$$\theta_f(\tau) = \frac{B}{1+B} + 6B\sum_{k=1}^{\infty} \exp(-\tau b_k^2)[b_k^2 B^2 + 9(1+B)]^{-1} \tag{7.4.7}$$

where the b_k are the non-zero roots of

$$\tan b = 3b/(3+Bb^2) \quad . \tag{7.4.8}$$

For this problem, we take

$$B = 1/2,$$

and for this value of B, the first 15 non-zero roots are given in Table 7.4.1.

TABLE 7.4.1. First 15 Non-zero Roots

3.972021016717	6.938670961459	9.942061135575
12.98602738139	16.05806834831	19.14860439986
22.25151701953	25.36296489242	28.48048660982
31.60245747142	34.72777199596	37.85565636325
40.98555486792	44.1170592249	47.24986331196

The roots shown in Table 7.4.1 were obtained by fixed point iteration on Eq. (7.4.8), and they were also calculated using Newton iterations. When these roots are used in (7.4.8), the residuals are less than 1.0-8 in all cases. These roots were used in Eqs. (7.4.6) and (7.4.7), and for $\tau \geq .04$ the partial sums were unchanged in the first 10 significant digits when using 10 or 15 terms in these series. From these circumstances, we infer that the first 15 terms will provide solutions accurate to at least eight significant places when $\tau \geq 0.04$.

For this problem we used the following input data in the namelists.

GRID

```
     KR=6,KZ=1,
     CØNTR=5,
```

Here we are using a smooth quintic B-spline.

```
     DELTA=2,
(Spherical geometry)
     NQR=6,NQZ=1,
     NMR=22,
     RMESH = 0.1, 0.2, 0.3, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.725,
             0.75, 0.775, 0.8, 0.825, 0.85, 0.875, 0.9, 0.925, 0.95,
             0.975, 0.985,
```

Here we are using a non-uniform mesh since the temperature is initially one at the surface (ξ=1) and zero for $0 \leq \xi < 1$. The temperature profile will initially have a large gradient near ξ=1, and the gradient will always be zero at ξ=0.

```
     INITSW=F, GUESSW=F, STEDSW=F, TRANSW=T,
```

Since the initial temperature is zero inside the sphere and one at the surface, the least squares fit, which INITSW=T, would provide would generate a very poor fit to this data. For this reason, the initial spline coefficients will be provided in namelist DATA.

```
     IRGRD=6,
     RGRID=0.2,0.4,0.6,0.8,0.9,1.0,
     IANAL=T,
```

The series solution as given in Eqs. (7.4.6) and (7.4.7) will be provided in the user subroutine ANAL.

```
     ALGBCS=F,
```

We are using the differential boundary condition option of the code.

DATA

```
NS3(1)=1,
ALPHA(1,3)=1.0, BETA(1,3)=0.0, GAMMA(1,3)=1.0,
NUTØUT=6,
UTØUT=0.0,0.04,0.08,0.12,0.16,0.2,
W=27*0.0,
```

Here we are providing the initial spline coefficients. Recall that the initial temperature is identically zero for $0 \leq \xi < 1$ and one for $\xi=1$. From the properties of the B-splines (cf. section 2.2) we can infer that if the dimension of the problem is N (the number of basis functions), then the initial spline coefficients W will satisfy

$W(j) \equiv 0$ for $1 \leq j \leq N-1$, and
$W(N) = 1.$

Since the default value for the initial spline coefficients is one, setting W=27*0.0, will provide this initial distribution for the spline coefficients. Of course, there is the problem of determining that N=28 for this problem. Recall from section 2 that for a one-dimensional problem in r

$N = N_r = KR+NMR\cdot(KR-CØNTR).$

For this problem KR=6, NMR=22, and CØNTR=KR-1=5; hence N=28.

```
EPS=1.D-6, HINIT=1.0-8,
GRAPH=T,
```

The user routines for this problem are characterized as follows.

RHØCP

```
RC=1.DO
```

DIFUSE

```
DIFUR=1.DO
DIFUZ=0.DO
```

VEL

```
VELR=0.DO
VELZ=0.DO
```

EXTSRC

```
VV=0.DO
```

FDEXTU

```
UU(1)=0.DO
UUR(1)=0.DO
UUZ(1)=0.DO
```

BRHØDT

As mentioned before, the boundary conditions for this problem are presented in differential form; hence we use the differential boundary condition option. This implies that we use subroutine BRHØDT rather than BRHØ. In BRHØDT, we return time differentiated boundary values on sides that have essential boundary conditions (where β=0), and we return undifferentiated boundary values on sides that have non-essential boundary conditions. In this problem we have $\frac{\partial \theta_s}{\partial \xi} = 0$ on side 1; this is a non-essential boundary condition, hence we return RHØV=0.DO on side 1. On side 3, we have $\theta_s = \theta_f$ where $\frac{\partial \theta_s}{\partial \tau} = -3/B \frac{\partial \theta_s}{\partial \xi}$. This is an essential boundary condition; hence we return the time derivative of θ_s on side 3, that is

```
RHØV = -3.DO*SPDENX(1)/B .
```

ANAL

In this routine we implement the series solutions given in Eqs. (7.4.6) and (7.4.7) using the 15 non-zero roots given in Table 7.4.1. For very small values of τ (in particular for $\tau = 0$) these series are very slowly converging, and 15 terms is not adequate to obtain a reasonable solution. For $\tau \geq 0.04$, 15 terms is more than adequate to obtain eight significant digits of accuracy in these series solutions.

In all these sample problems, we have used NQR=KR and NQZ=KZ. This is not necessarily the optimal choice; however, we have used these values to insure that errors in the approximation were due solely to the spatial approximation. This is also why we have used rather tight convergence criteria (EPS) in the ODE solver. In Table 7.4.2 we show the effect of varying the quadrature order for the case of a cubic B-spline approximation to this problem.

Using KR=4 (cubic B-splines) we ran this problem with NQR=3,4,6. Table 7.4.2 presents the solution values at t = 0.04 and ξ = 0.1 and ξ = 1.0.

TABLE 7.4.2. Effect of Quadrature Order on Approximation Error

NQR	$\xi = 0.1$	$\xi = 1.0$	CPU time(sec)
3	0.016126856864	0.4516148560	36
4	0.01612406559	0.4515455554	40
6	0.01612406537	0.4515455545	43
Series Solution	0.0158142829	0.44352229025	

The data shows that for this problem the choice NQR = KR-1 = 3 is the optimal choice. This is in agreement with the discussion for elliptic problems in [4]. Although the CPU times are not drastically different, it should be remembered that for two-dimensional problems the effect of using $(K-1)^2$ points per rectangle rather than K^2 points will have a significant effect on the running time.

```
STORAGE MAXIMA FOR THIS COMPILATION :
MAXBRK          30
MAXSP            6
MAXTQD           6
MAXK             6
MXNRNZ         100
MXNVAR         300
MAXNOT          40
MXRGRD          20
MXZGRD          20
```

```
 READING NAMELIST  GRID

KR =  6    KZ =  1
NTIR =  0 NTIZ =  0
NSPEC =  1
RLOW =    0.0                                RUP =    0.1000000000000000D+01
ZLOW =    0.0                                ZUP =    0.1000000000000000D+01
NMR = 22     NMZ =  0
 INITIAL CONTR = 13          INITIAL CONTZ = 13
GEOMETRY INDICATOR =  2
ADDITIONAL NON-INTERFACE R MESH POINTS
I =  1 RMESH(I) =    0.1000000000000000D+00
I =  2 RMESH(I) =    0.2000000000000000D+00
I =  3 RMESH(I) =    0.3000000000000000D+00
I =  4 RMESH(I) =    0.4000000000000000D+00
I =  5 RMESH(I) =    0.4500000000000000D+00
I =  6 RMESH(I) =    0.5000000000000000D+00
I =  7 RMESH(I) =    0.5500000000000000D+00
I =  8 RMESH(I) =    0.6000000000000000D+00
I =  9 RMESH(I) =    0.6500000000000000D+00
I = 10 RMESH(I) =    0.7000000000000000D+00
I = 11 RMESH(I) =    0.7250000000000000D+00
I = 12 RMESH(I) =    0.7500000000000000D+00
I = 13 RMESH(I) =    0.7750000000000000D+00
I = 14 RMESH(I) =    0.8000000000000000D+00
I = 15 RMESH(I) =    0.8250000000000000D+00
I = 16 RMESH(I) =    0.8500000000000000D+00
I = 17 RMESH(I) =    0.8750000000000000D+00
I = 18 RMESH(I) =    0.9000000000000000D+00
I = 19 RMESH(I) =    0.9250000000000000D+00
I = 20 RMESH(I) =    0.9500000000000000D+00
I = 21 RMESH(I) =    0.9750000000000000D+00
I = 22 RMESH(I) =    0.9850000000000000D+00
NO ADDITIONAL NON-INTERFACE Z MESH POINTS
QUADRATURE ORDER FOR R DIRECTION     6
QUADRATURE ORDER FOR Z DIRECTION     1
MATERIAL TABLE IS GIVEN AS MATL(RINDEX,ZINDEX)
MATERIAL TABLE FOR ZINDEX =    1
 1
CONSRV=  F
ALGRCS=  F
LOGICAL SWITCHES TO CONTROL PROGRAM
STEDSW =  F      GUESSW =  F      TRANSW =  T      INITSW =  F
ISTOPS =  F          ITRARS =  F          IANAL =  T
DUMPSW =  F
IPEVLA( 1) =  F
ORDER OF SPLINE DERIVATIVES COMPUTED IS  0
NUMBER OF USER SUPPLIED POINTS IN R DIRECTION  6
RGRID( 1) =        0.20000000D+00
RGRID( 2) =        0.40000000D+00
RGRID( 3) =        0.60000000D+00
RGRID( 4) =        0.80000000D+00
RGRID( 5) =        0.90000000D+00
RGRID( 6) =        0.10000000D+01
NUMBER OF USER SUPPLIED POINTS IN Z DIRECTION  1
ZGRID( 1) =        0.50000000D+00


 READING NAMELIST  DATA
```

```
CONTR TOO HIGH, BEING RESET TO       6-1

CONTZ TOO HIGH, BEING RESET TO       1-1

NVGAP =  0      NHGAP =  0      LR = 23      LZ =  1      NP = 28      NZ =  1

IL(I) =
 6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28

JL(J) =
 1

IREF(I) =
 1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1

JREF(J) =  1
MLTAB( 1, 1) =  1
MLTAB( 2, 1) =  1
MLTAB( 3, 1) =  1
MLTAB( 4, 1) =  1
MLTAB( 5, 1) =  1
MLTAB( 6, 1) =  1
MLTAB( 7, 1) =  1
MLTAB( 8, 1) =  1
MLTAB( 9, 1) =  1
MLTAB(10, 1) =  1
MLTAB(11, 1) =  1
MLTAB(12, 1) =  1
MLTAB(13, 1) =  1
MLTAB(14, 1) =  1
MLTAB(15, 1) =  1
MLTAB(16, 1) =  1
MLTAB(17, 1) =  1
MLTAB(18, 1) =  1
MLTAB(19, 1) =  1
MLTAB(20, 1) =  1
MLTAB(21, 1) =  1
MLTAB(22, 1) =  1
MLTAB(23, 1) =  1

HORIZONTAL ORDERING



NI=   1          NJ=  28          NCC= -29          MBW=   5



NIH=   1          NJH=  28          NCCH= -28          DM=   5


SYSTEM SIZE FOR THIS CASE
LP =    23      LZ =     1
NP =    28      NZ =     1
NVAR =   28

 THIS IS THE DIRECT VERSION
```

```
 THIS VERSION DOES NOT REQUIRE BOUNDARY CONDITIONS ON EVERY SIDE
FOR SPECIES NO.  1
 SIDE  1 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  2 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  3 ALPHA =   0.10000000D+01     BETA =   0.0                 GAMMA =   0.10000000D+01
 SIDE  4 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  0     NS2=  0     NS3=  1     NS4=  0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
SPECIES NO.  1 MATERIAL INDEX  1 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
SPECIES NO.  1 MATERIAL INDEX  1 HU2 =    0.10000000000000000D+01      HU4 =    0.10000000000000000D+01
REACTION RATES
FIRST ORDER RATES
CK INTO  1 FROM  1 IS    0.0
SECOND ORDER REACTION RATES ARE
CKK INTO K =  1 FOR KP =  1 INTO KPP =  1          CKK( 1, 1, 1) =    0.0

ISTDFQ = 100
TIME AND SPACE GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  6
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL 1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES
   0.0                    0.40000000000000000D-01   0.80000000000000000D-01   0.12000000000000000D+00
   0.16000000000000000D+00   0.20000000000000000D+00
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  T
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
EPS =    0.10000000000000000D-05      HINIT =    0.10000000000000000D-07      MF =  21      MXGORD =  5
CONTINUITY FOR R AND Z DIRECTIONS MAY HAVE BEEN RESET
CONTR =  5          CONTZ =  0
DEFAULT INITIAL COEFFICIENTS

        0.0                    0.0                    0.0                    0.0
        0.0                    0.0                    0.0                    0.0
        0.0                    0.0                    0.0                    0.0
        0.0                    0.0                    0.0                    0.0
        0.0                    0.0                    0.0                    0.0
        0.0                    0.0                    0.0                    0.0
        0.0                    0.0                    0.0                    0.10000000000000000D+01
```

```
READING NAMELIST  DATA
 CHANGES IN NAMELIST DATA MAY HAVE BEEN MADE FOR TRANSIENT
FOR SPECIES NO.  1
 SIDE  1 ALPHA =    0.0                 BETA =   -0.10000000D+01      GAMMA =    0.0
 SIDE  2 ALPHA =    0.0                 BETA =   -0.10000000D+01      GAMMA =    0.0
 SIDE  3 ALPHA =    0.10000000D+01      BETA =    0.0                 GAMMA =    0.10000000D+01
 SIDE  4 ALPHA =    0.0                 BETA =    0.10000000D+01      GAMMA =    0.0

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  0     NS2=  0     NS3=  1     NS4=  0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
SPECIES NO.  1 MATERIAL INDEX  1 HU1 =     0.10000000000000000D+01      HU3 =     0.10000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
SPECIES NO.  1 MATERIAL INDEX  1 HU2 =     0.10000000000000000D+01      HU4 =     0.10000000000000000D+01
REACTION RATES
FIRST ORDER RATES
CK INTO  1 FROM  1 IS     0.0
SECOND ORDER REACTION RATES ARE
CKK INTO K =  1 FOR KP =  1 INTO KPP =  1          CKK( 1, 1, 1) =     0.0

ISTDFO = 100
TIME AND SPACE GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  6
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL 1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES
   0.0                      0.40000000000000000D-01   0.80000000000000000D-01   0.12000000000000000D+00
   0.16000000000000000D+00  0.20000000000000000D+00
DATASET CREATED FOR USR IN GRAPHICS
 GRAPH =  T
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
EPS =    0.10000000000000000D-05      HINIT =    0.10000000000000000D-07      MF =  21     MXGORD =  5
CONTINUITY FOR R AND Z DIRECTIONS
CONTR =  5          CONTZ =  0
INITIAL COEFFICIENTS FOR TRANSIENT

        0.0                      0.0                      0.0                      0.0
        0.0                      0.0                      0.0                      0.0
        0.0                      0.0                      0.0                      0.0
        0.0                      0.0                      0.0                      0.0
        0.0                      0.0                      0.0                      0.0
        0.0                      0.0                      0.0                      0.0
        0.0                      0.0                      0.0                      0.10000000000000000D+01
```

```
 BEGIN TRANSIENT SOLUTION


PROUT FOR TIME =        0.0

  W =
      0.0                   0.0                   0.0                   0.0                   0.0                   0.0
      0.0                   0.0                   0.0                   0.0                   0.0                   0.0
      0.0                   0.0                   0.0                   0.0                   0.0                   0.0
      0.0                   0.0                   0.0                   0.0                   0.0                   0.0
      0.0                   0.0                   0.0                   0.10000000D+01
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.0                   0.0                   0.0                   0.0                   0.0
   0.10000000000000000D+01
ANALYTIC SOLN.
   0.3628622427957043D-01  -0.4014236151701241D-01   0.4964908689397154D-01  -0.7943315450942572D-01  -0.1639458956046248D-01
   0.9222987265908287D+00
MAXIMUM ERROR ON GRID IS        0.79433155D-01
   TOUT=    0.0


PROUT FOR TIME =        0.40000000D-01

  W =
      0.77644322D-02        0.77643167D-02        0.81191808D-02        0.97158559D-02        0.14324398D-01        0.25643782D-01
      0.43541572D-01        0.67325748D-01        0.94503114D-01        0.12196681D+00        0.15423085D+00        0.18721603D+00
      0.21899806D+00        0.24807101D+00        0.27348497D+00        0.29476734D+00        0.31591126D+00        0.33666278D+00
      0.35676236D+00        0.37595184D+00        0.39398124D+00        0.41061577D+00        0.42383937D+00        0.43464191D+00
      0.44188270D+00        0.44631052D+00        0.44842282D+00        0.44965329D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.1604262139991134D-01   0.5474224733076071D-01   0.1553235982774029D+00   0.3158131229674054D+00   0.3936327054263891D+00
   0.4496532880204012D+00
ANALYTIC SOLN.
   0.1581428292838039D-01   0.5397693116337743D-01   0.1531794171577383D+00   0.3114879115746942D+00   0.3882562087821096D+00
   0.4435222902504340D+00
MAXIMUM ERROR ON GRID IS        0.61309978D-02
   TOUT=    0.40000000000000000D-01  DELTA T =    0.40000000000000000D-01

ESTIMATED TIME FOR A CALL TO DRIVE        0.22900000D+04
TIME LEFT        0.16910000D+05


PROUT FOR TIME =        0.80000000D-01

  W =
      0.10065042D+00        0.10065040D+00        0.10160382D+00        0.11589373D+00        0.11731800D+00        0.13851348D+00
      0.16333127D+00        0.18883582D+00        0.21290343D+00        0.23381385D+00        0.25543126D+00        0.27499878D+00
      0.29211578D+00        0.30665237D+00        0.31866590D+00        0.32831529D+00        0.33754082D+00        0.34628367D+00
      0.35448903D+00        0.36210682D+00        0.36909243D+00        0.37540738D+00        0.38034638D+00        0.38434194D+00
      0.38700522D+00        0.38863099D+00        0.38940630D+00        0.38985795D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )
```

```
APPROXIMATE SOLUTION
   0.1196314310998135D+00    0.1744873050191648D+00    0.2554620235054770D+00    0.3374201818108623D+00    0.3689248227131347D+00
   0.3898579466767527D+00
ANALYTIC SOLN.
   0.1179822202598139D+00    0.1722877060870651D+00    0.2519620574092851D+00    0.3328071737814856D+00    0.3638853033719295D+00
   0.3845349888526385D+00
MAXIMUM ERROR ON GRID IS        0.53229578D-02
   TOUT=    0.8000000000000000D-01  DELTA T =    0.4000000000000000D-01

ESTIMATED TIME FOR A CALL TO DRIVE        0.23600000D+03
TIME LEFT        0.16674000D+05



PROUT FOR TIME =        0.12000000D+00

  W =
      0.20087741D+00       0.20087739D+00       0.20155334D+00       0.20459498D+00       0.21260373D+00       0.22685830D+00
      0.24274138D+00       0.25832591D+00       0.27242131D+00       0.28441242D+00       0.29639500D+00       0.30694360D+00
      0.31596457D+00       0.32349018D+00       0.32962449D+00       0.33450046D+00       0.33911715D+00       0.34345331D+00
      0.34748985D+00       0.35121011D+00       0.35459990D+00       0.35764770D+00       0.36002103D+00       0.36193592D+00
      0.36321032D+00       0.36398786D+00       0.36435861D+00       0.36457457D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.2140566246308213D+00    0.2496016417181893D+00    0.2963295827781134D+00    0.3390470371101609D+00    0.3545144271750798D+00
   0.3645745727106807D+00
ANALYTIC SOLN.
   0.2111233502251016D+00    0.2461833001422379D+00    0.2922749938292047D+00    0.3344125002250399D+00    0.3496703268585085D+00
   0.3595946081811449D+00
MAXIMUM ERROR ON GRID IS        0.49799645D-02
   TOUT=    0.1200000000000000D+00  DELTA T =    0.3999999999999999D-01

ESTIMATED TIME FOR A CALL TO DRIVE        0.12700000D+03
TIME LEFT        0.16547000D+05



PROUT FOR TIME =        0.16000000D+00

  W =
      0.26341068D+00       0.26341067D+00       0.26379612D+00       0.26553061D+00       0.27008624D+00       0.27811972D+00
      0.28696889D+00       0.29555614D+00       0.30324980D+00       0.30974522D+00       0.31618704D+00       0.32181629D+00
      0.32660100D+00       0.33057313D+00       0.33379856D+00       0.33635486D+00       0.33876861D+00       0.34102992D+00
      0.34313008D+00       0.34506163D+00       0.34681836D+00       0.34839540D+00       0.34962191D+00       0.35061078D+00
      0.35126864D+00       0.35166997D+00       0.35186133D+00       0.35197281D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.2708917563317534D+00    0.2907383433043359D+00    0.3161403307985439D+00    0.3387305105100299D+00    0.3467734494822907D+00
   0.3519728069533007D+00
ANALYTIC SOLN.
   0.2671846397385692D+00    0.2867605575634516D+00    0.3118169842213935D+00    0.3341008376133969D+00    0.3420350294256288D+00
   0.3471639976483230D+00
MAXIMUM ERROR ON GRID IS        0.48088093D-02
   TOUT=    0.1600000000000000D+00  DELTA T =    0.4000000000000001D-01

ESTIMATED TIME FOR A CALL TO DRIVE        0.15500000D+03
TIME LEFT        0.16392000D+05
```

```
PROUT FOR TIME =        0.20000000D+00

  W =
      0.29805915D+00       0.29835914D+00       0.29826799D+00       0.29920778D+00       0.30167454D+00       0.30601416D+00
      0.31078024D+00       0.31539188D+00       0.31951332D+00       0.32298580D+00       0.32642263D+00       0.32941992D+00
      0.33196329D+00       0.33407188D+00       0.33576228D+00       0.33713675D+00       0.33841471D+00       0.33961111D+00
      0.34072152D+00       0.34174218D+00       0.34266999D+00       0.34350250D+00       0.34414974D+00       0.34467144D+00
      0.34501846D+00       0.34523015D+00       0.34533109D+00       0.34538989D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA TIMEX )

APPROXIMATE SOLUTION
   0.30210783033975780+00   0.31280305920223820+00   0.32639604060076030+00   0.33839432473305220+00   0.34264617105957000+00
   0.34538988525446780+00
ANALYTIC SOLN.
   0.29797588653427880+00   0.30852568588774700+00   0.32193401202605910+00   0.33376975389267480+00   0.33796423018026670+00
   0.34067100752019700+00
MAXIMUM ERROR ON GRID IS        0.47188777D-02
   TOUT=    0.20000000000000000+00  DELTA T =    0.39999999999999990-01

ESTIMATED TIME FOR A CALL TO DRIVE        0.30800000D+03
TIME LEFT        0.16084000D+05

****** NORMAL DUMP AT END OF TIMEX
         I =      6           TOUT =     0.20000000000000000+00


END OF CASE
```

READING NAMELIST GRID

7.5 Heat Conduction with Chemical Heat Source

This problem involves a simple model for a fixed-bed flow reactor. The problem will illustrate the use of interface conditions and the use of material tables. This problem is taken from [8] where a solution to the problem is also given.

The reactor extends from $Z = -\infty$ to $Z = +\infty$ and is divided into three zones with the central zone ($0 < z < L$) being the reaction zone. Radial velocity gradients are neglected and the walls are well insulated so that the temperature is independent of the radial coordinate r. The task is to find the steady-state axial temperature distribution $T(z)$ when the fluid enters at $z = -\infty$ with a uniform temperature T_1 and are average linear velocity without packing of $v_1 = w/\pi R^2 \rho_1$ where R is the radius of the reactor and ρ_1 is the fluid density. The volume rate of thermal energy production by chemical reactions S_c is assumed to be of the form

$$S_c(z) = S_{c1}\left(\frac{T(z) - T^0}{T_1 - T^0}\right)$$

where T^0 and S_{c1} are given constants.

The governing equations are as follows.

$$\rho_1 v_1 C_p \frac{dT}{dz} = k \frac{d^2T}{dz^2} \quad \text{for} \quad z < 0,$$

$$\rho_1 v_1 C_p \frac{dT}{dz} = k \frac{d^2T}{dz^2} + S_{c1}\left(\frac{T - T^0}{T_1 - T^0}\right) \quad \text{for} \quad 0 < z < L, \tag{7.5.1}$$

$$\rho_1 v_1 C_p \frac{dT}{dz} = k \frac{d^2T}{dz^2} \quad \text{for} \quad L < z .$$

The following boundary and interface conditions are used.

$$T = T_1 \quad \text{at} \quad z = -\infty$$

$$T(0^-) = T(0^+)$$

$$k \left.\frac{dT}{dz}\right|_{0^-} = k \left.\frac{dT}{dz}\right|_{0^+}$$

$$T(L^-) = T(L^+) \tag{7.5.2}$$

$$k \left.\frac{dT}{dz}\right|_{L^-} = k \left.\frac{dT}{dz}\right|_{L^+}$$

$$\frac{dT}{dz} = 0 \quad \text{at} \quad z = +\infty$$

In [8] the solution to this problem is given in terms of dimensionless variables. Thus

$$\zeta = z/L, \quad \theta = (T-T^0)/(T_1-T^0), \quad B = \rho_1 v_1 C_p L/k, \quad \text{and}$$

$$F = S_{c1} L/\rho_1 v_1 C_p (T_1-T^0).$$

The governing equations are then of the following form:

$$\frac{d\theta}{d\zeta} = \frac{1}{B}\frac{d^2\theta}{d\zeta^2} \quad \text{for} \quad \zeta < 0,$$

$$\frac{d\theta}{d\zeta} = \frac{1}{B}\frac{d^2\theta}{d\zeta^2} + \theta F \quad \text{for} \quad 0 < \zeta < 1, \tag{7.5.3}$$

$$\frac{d\theta}{d\zeta} = \frac{1}{B}\frac{d^2\theta}{d\zeta^2} \quad \text{for} \quad 1 < \zeta.$$

The boundary conditions and interface conditions have the form:

$$\theta = 1 \quad \text{at} \quad \zeta = -\infty$$

$$\theta(0^-) = \theta(0^+)$$

$$\frac{1}{B}\frac{d\theta}{d\zeta}\bigg|_{0^-} = \frac{1}{B}\frac{d\theta}{d\zeta}\bigg|_{0^+}$$

(7.5.4)
$$\theta(1^-) = \theta(1^+)$$

$$\frac{1}{B}\frac{d\theta}{d\zeta}\bigg|_{1^-} = \frac{1}{B}\frac{d\theta}{d\zeta}\bigg|_{1^+}$$

$$\frac{d\theta}{d\zeta} = 0 \quad \text{at} \quad \zeta = +\infty.$$

The solution, from [8], is then given by the following expressions. Let

$$m_3 = \frac{1}{2}B(1 - \sqrt{1 - 4F/B})$$

$$m_4 = \frac{1}{2}B(1 + \sqrt{1 - 4F/B})$$

Assume that $1 - 4F/B > 0$ and set

$E = m_4^2\exp(m_4) - m_3^2\exp(m_3)$, then

(7.5.5)
$$\begin{cases} \theta(\zeta) = 1 + \frac{1}{E}[m_3 m_4(\exp(m_4)-\exp(m_3))]\exp[(m_3+m_4)\zeta] & \text{for } \zeta < 0, \\ \theta(\zeta) = \frac{1}{E}[m_4\exp(m_4+m_3\zeta) - m_3\exp(m_3+m_4\zeta)](m_3+m_4) & \text{for } 0 < \zeta < 1, \\ \theta(\zeta) = \frac{1}{E}(m_4^2-m_3^2)\exp(m_3+m_4) & \text{for } 1 < \zeta. \end{cases}$$

This problem was run with B=8 and F=-1, and the following data was used in the namelists.

Grid

KR=1,KZ=4,

Since the problem is one dimensional and presented as a problem in z, we use this coordinate.

NTIZ=2,

This is the number of interfaces for this problem.

IFTYPZ=1,1,

This vector indicates that the two interfaces are of type 1; that is, we require that the approximation be continuous at these interfaces. For this particular problem the diffusivity is the same in each of the three sections; thus the solution given in Eq. (7.5.5) has continuous first derivative at these interfaces. Using interfaces, our approximations will not have a continuous first derivative at these interfaces; although the discrepancy will tend to zero as the mesh or spline order is increased. If we had wished to use splines with continuous first derivatives at these interfaces, we would have avoided using interfaces (i.e. we would have set NTIZ=0,) and instead we would have used the variable continuity index INUZ to specify the continuity desired at those interfaces. However, in this problem we wish to illustrate the use of material regions and interfaces.

ZIF=0.0,1.0,

These are the coordinates of the interfaces.

ZLØW=-2.0,

At z=-2, the analytic solution is equal to 1 to 7 decimal places; hence the error due to the position of the boundary should not contaminate the approximation error.

ZUP=1.2,

Since the solution is flat in the interval [1,∞) this choice is appropriate. Thus for numerical purposes, the reactor is in the interval [-2,1.2].

NMZ=11,

ZMESH=-1.5,-1.0,-0.8,-0.6,-0.4,-0.2,0.2,0.4,0.6,0.8,1.1,

Note that these additional mesh points do not include the interface points. Thus the total number of mesh points will be 13.

NQR=1,NQZ=4,

MATL(1,1)=1, MATL(1,2)=2, MATL(1,3)=3,

Here we assign material indices to each of the three sections. Material #1 is assigned to the interval [-2.0,0), material #2 is assigned to (0,1), and material #3 is assigned to (1.0,1.2). Recall that the vector ZIF subdivided the interval [-2.0,1.2] into three

INITSW=T, GUESSW=F, STEDSW=T, TRANSW=F,

ISTDFQ=100,

This is a steady-state calculation, so we are only interested in the final solution. However, ISTDFQ=100, means that we will get output every 100 time steps. This is useful as a check on how the calculation is proceeding. When the problem finishes, i.e. when a steady-state has been reached, the program will produce the printed output regardless of the value of ISTDFQ. Thus we could have set it to 1,000 or some large number in order to suppress the intermediate output.

JZGRD=6,

ZGRID=-0.3,0.0,0.2,0.6,1.0,1.1,

Here we are asking for output at these six locations.

IANAL=T,

We have an analytic solution for this problem so we will provide this solution.

Namelist DATA

NS2(1)=1,

The boundary conditions on sides 1, 3, and 4 are non-essential which is the default value.

ALPHA(1,2)=1.0, BETA(1,2)=0.0, GAMMA(1,2)=1.0,

EPS=1.D-5, HINIT=1.D-5,

GRAPH=T,

This completes the input for the namelists. The user-supplied subroutines used the following data.

RHØCP

RC=1.DO

DIFUSE

DIFUR=0.DO
DIFUZ=1.DO/8.DO

VEL

VELR=0.DO
VELZ=1.DO

EXTSRC

Recall that only the central section has a non-zero source. This central section had material index 2; so we can use this material index in this routine.

```
VV=0.DO
IF(IMATL .EQ. 2) VV=-SPDEN(1)
```

FDEXTU

The source is non-zero only in the central sections where it is equal to -SPDEN(1). Thus UU(1), UUR(1), and UUZ(1) are zero except in the central section where we use:

```
IF(IMATL .EQ. 2) UU(1)=-1.DO
```

INDATA

Any reasonable estimate will do; here we use

```
UU=1.DO
```

BRHØ

The boundary conditions are taken as $\theta(-2)=1$ and $\theta'(1.2)=0$. From Eq. (7.5.5), we see that the solution is constant for $z > 1$; hence the choice ZUP=1.2 is reasonable and leads to no errors in the approximation. The choice of ZLØW=-2 as an approximation to $-\infty$ does lead to an error in the approximation; however, the solution at z = -2 is equal to 1 to 7 decimal places; hence we do not expect this choice for ZLØW to cause any significant error. Thus on side 2 we use:

```
RHØV=1.DO
```

and on side 4 we use

```
RHØV=0.DO
```

ANAL

In this routine we implement Eq. (7.5.5).

The printed output corresponding to this data is given on the following pages. In addition, for this problem we also give some indication of the effect of spline order and mesh size on the approximation error. For a fixed mesh size NMZ=11 we used splines of order 2, 3, and 4. Also for a spline order of 4 we halved the mesh size (NMZ=21) in the interval [-1.0,1.0]. We sampled the maximum error over the points -0.3, 0.0, 0.2, 0.6, 1.0, and 1.1;

in all cases the maximum error was at -0.3. In the following table we give the error at the first interface 0.0, at z = 0.2, and the maximum error (which occurred at z = -0.3).

TABLE 7.5.1. Approximation Errors

KZ	z = 0 (interface)	z = 0.2	Max. Error (z = -0.3)
2	2194.65	1892.3	2793.09
3	0.795	32.92	765.938
4	1.609	12.28	30.767
4 NMZ=21	0.485	0.551	3.527

In considering this data we must bear in mind that the time integration error control parameter (EPS) was only 1.D-5 for these runs; thus there could be some contamination in the results for k=4. In addition, the fact that the maximum error was always at -0.3 may indicate that the position of the mesh in the first section is contributing to the error. (Recall that we have used z = -2.0,-1.5,-1.0,... as mesh points, and the solution at z = -1.0 is 1 to 3 decimal places; thus these mesh positions may be contaminating the error when k = 4.) In any case the results show substantial reductions in the error at least from k = 2 to k = 3, and for k = 4 we see a behavior of $\epsilon = O(h^4)$ when refining the mesh.

```
STORAGE MAXIMA FOR THIS COMPILATION :
MAXBRK          30
MAXSP            6
MAXTOD           6
MAXK             6
MXNRNZ         100
MXNVAR         300
MAXNOT          40
MXRGRD          20
MXZGRD          20
```

```
 READING NAMELIST  GRID

KR =  1    KZ =  4
NTIR =  0 NTIZ =  2
NSPEC =  1
RLOW =    0.0                                RUP =    0.10000000000000000D+01
ZLOW =   -0.20000000000000000D+01            ZUP =    0.12000000000000000D+01
NMR =  0     NMZ =  11
 INITIAL CONTR = 13           INITIAL CONTZ = 13
GEOMETRY INDICATOR =  0
INTERFACE Z-DIRECTION MESH POINTS
I =  1     IFTYPZ(I) =  1     ZIF(I) =    0.0
I =  2     IFTYPZ(I) =  1     ZIF(I) =    0.10000000000000000D+01
NO ADDITIONAL NON-INTERFACE R MESH POINTS
ADDITIONAL NON-INTERFACE Z MESH POINTS
I =  1 ZMESH(I) =   -0.15000000000000000D+01
I =  2 ZMESH(I) =   -0.10000000000000000D+01
I =  3 ZMESH(I) =   -0.80000000000000000D+00
I =  4 ZMESH(I) =   -0.60000000000000000D+00
I =  5 ZMESH(I) =   -0.40000000000000000D+00
I =  6 ZMESH(I) =   -0.20000000000000000D+00
I =  7 ZMESH(I) =    0.20000000000000000D+00
I =  8 ZMESH(I) =    0.40000000000000000D+00
I =  9 ZMESH(I) =    0.60000000000000000D+00
I = 10 ZMESH(I) =    0.80000000000000000D+00
I = 11 ZMESH(I) =    0.11000000000000000D+01
QUADRATURE ORDER FOR R DIRECTION     1
QUADRATURE ORDER FOR Z DIRECTION     4
MATERIAL TABLE IS GIVEN AS MATL(RINDEX,ZINDEX)
MATERIAL TABLE FOR ZINDEX =   3
 3
MATERIAL TABLE FOR ZINDEX =   2
 2
MATERIAL TABLE FOR ZINDEX =   1
 1
CONSRV=  F
ALGBCS=  T
LOGICAL SWITCHES TO CONTROL PROGRAM
STEDSW =  T      GUESSW =  F      TRANSW =  F      INIISW =  T
ISTDRS =  F          ITRARS =  F          IANAL =  T
DUMPSW =  F
IREVLA( 1) =  F
ORDER OF SPLINE DERIVATIVES COMPUTED IS  0
NUMBER OF USER SUPPLIED POINTS IN R DIRECTION  1
RGRID( 1) =       0.50000000D+00
NUMBER OF USER SUPPLIED POINTS IN Z DIRECTION  6
ZGRID( 1) =      -0.30000000D+00
ZGRID( 2) =       0.0
ZGRID( 3) =       0.20000000D+00
ZGRID( 4) =       0.60000000D+00
ZGRID( 5) =       0.10000000D+01
ZGRID( 6) =       0.11000000D+01


 READING NAMELIST  DATA


CONTR TOO HIGH, BEING RESET TO      1-1
```

```
CONTZ TOO HIGH, BEING RESET TO       4-1

NVGAP =  0      NHGAP =  0      LR =  1      LZ = 14      NR =  1      NZ = 21

IL(I) =
 1

JL(J) =
 4   5   6   7   8   9  10  13  14  15  16  17  20  21

IREF(I) =
 1

JREF(J) =  1   1   1   1   1   1   1   2   2   2
2   2   3   3
MLTAB( 1, 1) =  1
MLTAB( 1, 2) =  1
MLTAB( 1, 3) =  1
MLTAB( 1, 4) =  1
MLTAB( 1, 5) =  1
MLTAB( 1, 6) =  1
MLTAB( 1, 7) =  1
MLTAB( 1, 8) =  2
MLTAB( 1, 9) =  2
MLTAB( 1,10) =  2
MLTAB( 1,11) =  2
MLTAB( 1,12) =  2
MLTAB( 1,13) =  3
MLTAB( 1,14) =  3

HORIZONTAL ORDERING



NI=   1          NJ=   1          NCC=  -2          MBW=   3



NIH=   1          NJH=   1          NCCH=  -1          DM=   3


SYSTEM SIZE FOR THIS CASE
LR =     1     LZ =    14
NR =     1     NZ =    21
NVAR =   21

 THIS IS THE DIRECT VERSION

 THIS VERSION DOES NOT REQUIRE BOUNDARY CONDITIONS ON EVERY SIDE
FOR SPECIES NO.  1
 SIDE  1 ALPHA =   0.0                BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  2 ALPHA =   0.10000000D+01     BETA =   0.0                 GAMMA =   0.10000000D+01
 SIDE  3 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0
 SIDE  4 ALPHA =   0.0                BETA =   0.10000000D+01      GAMMA =   0.0

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  0     NS2=  1     NS3=  0     NS4=  0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
```

```
SPECIES NO.  1 MATERIAL INDEX  1 HU1 =     0.1000000000000000D+01      HU3 =     0.1000000000000000D+01
SPECIES NO.  1 MATERIAL INDEX  2 HU1 =     0.1000000000000000D+01      HU3 =     0.1000000000000000D+01
SPECIES NO.  1 MATERIAL INDEX  3 HU1 =     0.1000000000000000D+01      HU3 =     0.1000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
SPECIES NO.  1 MATERIAL INDEX  1 HU2 =     0.1000000000000000D+01      HU4 =     0.1000000000000000D+01
FOR SPECIES  1 INTERFACES   1  1 HHGAP =     0.0
FOR SPECIES  1 INTERFACES   2  1 HHGAP =     0.0
REACTION RATES
FIRST ORDER RATES
CK INTO  1 FROM  1 IS    0.0
SECOND ORDER REACTION RATES ARE
CKK INTO K =  1 FOR KP =  1 INTO KPP =  1          CKK( 1, 1, 1) =    0.0

ISTDFQ = 100
TIME AND SPACE GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  6
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL 1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES
   0.0                      0.4000000000000000D-01   0.8000000000000000D-01   0.1200000000000000D+00
   0.1600000000000000D+00   0.2000000000000000D+00
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  F
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
EPS =    0.1000000000000000D-04      HINIT =    0.1000000000000000D-04      MF =  21     MXGORD =  5
CONTINUITY FOR R AND Z DIRECTIONS MAY HAVE BEEN RESET
CONTR =  0          CONTZ =  3
DEFAULT INITIAL COEFFICIENTS


        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01        0.1000000000000000D+01
        0.1000000000000000D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1000000000000000D+01
   0.9999999999999999D+00
   0.9999999999999995D+00
   0.9999999999999997D+00
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  2)-TH RECTANGLE.
(FROM INIFIT)
   0.9999999999999999D+00
   0.1000000000000000D+01
   0.1000000000000000D+01
   0.1000000000000000D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  3)-TH RECTANGLE.
(FROM INIFIT)
   0.1000000000000000D+01
   0.1000000000000000D+01
   0.1000000000000000D+01
   0.1000000000000000D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  4)-TH RECTANGLE.
(FROM INIFIT)
   0.9999999999999998D+00
   0.9999999999999998D+00
   0.1000000000000000D+01
```

```
   0.1000000000000000D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  5)-TH RECTANGLE.
(FROM INIFIT)
   0.9999999999999998D+00
   0.9999999999999998D+00
   0.9999999999999997D+00
   0.9999999999999998D+00
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  6)-TH RECTANGLE.
(FROM INIFIT)
   0.9999999999999996D+00
   0.9999999999999996D+00
   0.9999999999999997D+00
   0.9999999999999998D+00
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  7)-TH RECTANGLE.
(FROM INIFIT)
   0.9999999999999999D+00
   0.9999999999999998D+00
   0.9999999999999997D+00
   0.9999999999999996D+00
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  8)-TH RECTANGLE.
(FROM INIFIT)
   0.9999999999999997D+00
   0.9999999999999996D+00
   0.9999999999999996D+00
   0.9999999999999998D+00
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  9)-TH RECTANGLE.
(FROM INIFIT)
   0.1000000000000000D+01
   0.1000000000000000D+01
   0.1000000000000000D+01
   0.9999999999999998D+00
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1, 10)-TH RECTANGLE.
(FROM INIFIT)
   0.9999999999999994D+00
   0.9999999999999994D+00
   0.9999999999999997D+00
   0.1000000000000000D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1, 11)-TH RECTANGLE.
(FROM INIFIT)
   0.1000000000000000D+01
   0.1000000000000000D+01
   0.9999999999999998D+00
   0.9999999999999995D+00
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1, 12)-TH RECTANGLE.
(FROM INIFIT)
   0.1000000000000000D+01
   0.9999999999999991D+00
   0.9999999999999995D+00
   0.1000000000000000D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1, 13)-TH RECTANGLE.
(FROM INIFIT)
   0.1000000000000000D+01
   0.1000000000000000D+01
   0.9999999999999996D+00
   0.1000000000000000D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1, 14)-TH RECTANGLE.
(FROM INIFIT)
   0.1000000000000000D+01
   0.9999999999999996D+00
   0.1000000000000000D+01
   0.1000000000000000D+01
TINTV,T2,ICOR      0.70000000D+01      0.15596000D+05   22
```

```
                                REVISED ESTIMATE OF ITERATIONS POSSIBLE BEFORE DUMP  2206

                                        1 CALLS TO DRIVE HAVE OCCURRED


PROUT FOR TIME =      -0.10C00000D+05

  W =
      0.10000000D+01      0.99999997D+00      0.10000001D+01      0.99999992D+00      0.10000001D+01      0.99999966D+00
      0.10000007D+01      0.99999837D+00      0.10000030D+01      0.99999498D+00      0.99998708D+00      0.99999146D+00
      0.99998950D+00      0.99998997D+00      0.99999070D+00      0.99998830D+00      0.99999307D+00      0.10000054D+01
      0.99999491D+00      0.10000024D+01      0.99999911D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA STEADY)

APPROXIMATE SOLUTION
   0.9999994021873465D+00
ANALYTIC SOLN.
   0.4028316717141206D+00

APPROXIMATE SOLUTION
   0.9999930695433205D+00
ANALYTIC SOLN.
   0.4028316717141206D+00

APPROXIMATE SOLUTION
   0.9999900094408449D+00
ANALYTIC SOLN.
   0.5252514372701850D+00

APPROXIMATE SOLUTION
   0.9999900399945185D+00
ANALYTIC SOLN.
   0.7510744022932562D+00

APPROXIMATE SOLUTION
   0.9999949796209109D+00
ANALYTIC SOLN.
   0.8989850414801589D+00

APPROXIMATE SOLUTION
   0.9999996675609284D+00
ANALYTIC SOLN.
   0.9908361297114651D+00
MAXIMUM ERROR ON GRID IS       0.59716773D+00
 AT TOUT = -0.10000000D+05 SOBN =  0.12920189D+01 AND HUSED =  0.10000000D-04
NQUSED =  1

SUPN =
      0.12920189D-04

                                        94 CALLS TO DRIVE HAVE OCCURRED


PROUT FOR TIME =      -0.99906467D+04

  W =
```

```
      0.10000000D+01      0.99998093D+00      0.10000333D+01      0.99996400D+00      0.99994792D+00      0.99933442D+00
      0.99749753D+00      0.98620943D+00      0.95144504D+00      0.89898343D+00      0.84513152D+00      0.74697465D+00
      0.62426418D+00      0.52188192D+00      0.43953522D+00      0.40357934D+00      0.40283100D+00      0.40283096D+00
      0.40283085D+00      0.40283078D+00      0.40283078D+00
 VALUES OF CONCENTRATIONS U OF THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA STEADY)

APPROXIMATE SOLUTION
   0.40283085949473510+00
ANALYTIC SOLN.
   0.40283167171412060+00

APPROXIMATE SOLUTION
   0.40283100187932880+00
ANALYTIC SOLN.
   0.40283167171412060+00

APPROXIMATE SOLUTION
   0.52522117783164410+00
ANALYTIC SOLN.
   0.52525143727018500+00

APPROXIMATE SOLUTION
   0.75106212257460790+00
ANALYTIC SOLN.
   0.75107440229325620+00

APPROXIMATE SOLUTION
   0.89898343251789140+00
ANALYTIC SOLN.
   0.89898504148015890+00

APPROXIMATE SOLUTION
   0.99080536281195290+00
ANALYTIC SOLN.
   0.99083612971146510+00
MAXIMUM ERROR ON GRID IS       0.30766900D-04
 AT TOUT = -0.99906467D+04 SOSN =  0.71516391D-05 AND HUSED =  0.94847124D+00
NQUSED =  3

SUPN =
      0.24303424D-05      0.20642440D-05      0.44092682D-06
 PHASE 0 COMPLETE
```

```
 READING NAMELIST  DATA

 CHANGES IN NAMELIST DATA MAY HAVE BEEN MADE FOR TRANSIENT
FOR SPECIES NO.  1
 SIDE  1 ALPHA =   0.0                  BETA =  -0.10000000D+01      GAMMA =   0.0
 SIDE  2 ALPHA =   0.10000000D+01       BETA =   0.0                 GAMMA =   0.10000000D+01
 SIDE  3 ALPHA =   0.0                  BETA =   0.10000000D+01      GAMMA =   0.0
 SIDE  4 ALPHA =   0.0                  BETA =   0.10000000D+01      GAMMA =   0.0

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  0     NS2=  1     NS3=  0     NS4=  0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
SPECIES NO.  1 MATERIAL INDEX  1 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
SPECIES NO.  1 MATERIAL INDEX  2 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
SPECIES NO.  1 MATERIAL INDEX  3 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
SPECIES NO.  1 MATERIAL INDEX  1 HU2 =    0.10000000000000000D+01      HU4 =    0.10000000000000000D+01
FOR SPECIES  1 INTERFACES   1  1 HHGAP =    0.0
FOR SPECIES  1 INTERFACES   2  1 HHGAP =    0.0
REACTION RATES
FIRST ORDER RATES
CK INTO  1 FROM  1 IS    0.0
SECOND ORDER REACTION RATES ARE
CKK INTO K =  1 FOR KP =  1 INTO KPP =  1          CKK( 1, 1, 1) =    0.0

ISTDFQ = 100
TIME AND SPACE GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  6
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL 1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES
   0.0                    0.40000000000000000D-01   0.80000000000000000D-01   0.12000000000000000D+00
   0.16000000000000000D+00   0.20000000000000000D+00
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  F
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
EPS =    0.10000000000000000D-04      HINIT =    0.10000000000000000D-04      MF =  21     MXGORD =  5
CONTINUITY FOR R AND Z DIRECTIONS
CONTR =  0          CONTZ =  3
INITIAL COEFFICIENTS FOR TRANSIENT

       0.10000000000000000D+01      0.99998093409079710+00      0.10000332928341970+01      0.99996399881938990+00
       0.99994792125261730+00      0.99933441904245620+00      0.99749753173154610+00      0.98620943116953330+00
       0.95144504319415460+00      0.89898343251789140+00      0.84513152023646750+00      0.74697464950089850+00
       0.62426416183980200+00      0.52188191704370020+00      0.43953521697526240+00      0.40357933599796030+00
       0.40283100187932880+00      0.40283095504977880+00      0.40283084995813150+00      0.40283078301289900+00
       0.40283078285906720+00


END OF CASE
```

READING NAMELIST GRID

7.6 A Defect-Diffusion Problem in One-Dimensional Spherical Geometry

This problem involves the distribution of two reacting species in a domain bounded by two concentric spheres, i.e. $\Omega = \{r: R_L \leq r \leq R_U\}$. Let $u_i(r,t)$, i=1,2, denote the concentration of the i^{th} species subject to the following governing equations.

$$\begin{aligned} \frac{\partial u_1}{\partial t} &= D_V \Delta u_1 + D_R - C_{iv} u_1 u_2 - C_{sv} C_s (u_1 - VATV) \\ \frac{\partial u_2}{\partial t} &= D_I \Delta u_2 + D_R - C_{iv} u_1 u_2 - C_{si} C_s u_2 \end{aligned} \tag{7.6.1}$$

$$\begin{aligned} u_1(R_L,t) &= VEQL; \; u_1(R_u,t) = VATV \\ u_2(R_L,t) &= XIEQL; \; u_2(R_u,t) = XIEQL \end{aligned} \tag{7.6.2}$$

$$\begin{aligned} u_1(r,0) &\equiv VEQL \\ u_2(r,0) &\equiv XIEQL \end{aligned} \tag{7.6.3}$$

where Δ denotes the Laplacian in spherical geometry, and

$$\begin{aligned} R_L &= 10^{-6}, \quad R_U = 10^{-5} \\ D_V &\doteq 1.98354 \cdot 10^{-7} \\ D_I &\doteq 9.37700 \cdot 10^{-4} \\ D_R &= 10^{-3} \\ C_{iv} &\doteq 3.40606 \cdot 10^{13} \\ C_{sv} &\doteq 3.70750 \cdot 10^{8} \\ C_s &\doteq 7.38716 \cdot 10^{-8} \\ C_{si} &\doteq 1.75269 \cdot 10^{12} \\ VATV &\doteq 1.65843 \cdot 10^{-7} \\ VEQL &\doteq 1.35436 \cdot 10^{-7} \\ XIEQL &\doteq 6.75044 \cdot 10^{-18} \end{aligned} \tag{7.6.4}$$

For the numerical approximation of this problem, the following input parameters are used in the Namelist input.

Namelist GRID

DELTA=2,

Spherical geometry.

KR=3,

B-splines of degree 2 are used in the radial direction.

KZ=1,

B-splines of zero degree are used on this axis. The problem is one dimensional.

NSPEC=2,

Two species.

RLØW=1.0-6, RUP=1.D-5,

Inner and outer radii of the concentric spheres. Defines the spatial domain in the radial direction.

NMR=10,

Number of interior breakpoints not including the end points.

RMESH = 1.1D-6, 1.2D-6, 1.4D-6, 2.4D-6, 4.4D-6, 6.6D-6, 8.6D-6, 9.6D-6, 9.8D-6, 9.9D-6,

Location of the radial breakpoints.

TRANSW=F,

A steady-state calculation will be done. We will not use the transient option.

IRGRD=4,

Number of output points in the radial direction.

RGRID = 3.5D-6, 5.D-6, 8.D-6, 9.D-6,

Position of the radial output points.

NQR=3,
NQZ=1,

These values are the default values; however they are displayed here in order to call attention to the fact that this problem fails with NQR = KR-1 = 2 and NQZ = KZ-1 = 0.

Namelist DATA

DRCHLT(1,1)=T, DRCHLT(2,1)=T,

Dirichlet boundary conditions for both species on side 1 (the left

or inner boundary).

DRCHLT(1,3)=T, DRCHLT(2,3)=T,

Dirichlet boundary conditions for both species on side 3 (the right or outer boundary).

ISTDFQ=100,

Frequency of output for a steady-state calculation. Printout will occur every ISTDFQ time steps.

GRAPH=T,

Graph of the solution is desired.

EPS=1.D-3,

Local error tolerance for the time integration in the ODE solver.

HINIT=1.D-15,

Initial step size in time used in the ODE solver.

The user-supplied subroutines are discussed below.

MASTER DRIVER

The data for this problem is temperature dependent and is generated in the MASTER DRIVER. This case corresponds to a temperature of 900. The MASTER DRIVER is also used to allocate space for the arrays AL and PW with NAL=100 and NPW=420 for this case. With the use of quadratic B-splines over 11 intervals, we have 13 unknowns for each species for a total system size of 26. This gives a required storage for AL of 91 and for PW of 416. This data is printed in the output.

RHOCP

This subroutine provides the coefficients of the time derivatives in Equation (7.6.1). Since these coefficients are equal to one, this routine returns 1.D0.

VEL

This subroutine returns the coefficients of the convective terms in Equation (7.6.1). Since there are no convective terms, this routine returns 0.D0.

DIFUSE

This routine provides the diffusion coefficients D_V and D_I. Note that the user does not concern himself with the form of the Laplacian in spherical geometry. This is taken care of in Namelist GRID by

setting DELTA=2,. Note also that since this problem is one dimensional (all quantities depend only on the R-coordinate), this routine sets the Z-component of the diffusion coefficient to 0.D0.

EXTSRC

This routine returns the source terms $s_1 = D_R - C_{iv}u_1u_2 - C_{sv}C_s(u_1 - VATV)$ for species 1, and $s_2 = D_R - C_{iv}u_1u_2 - C_{si}C_s u_2$ for species 2.

FDEXTU

This routine provides the Frechet derivatives (or an approximation thereof) of the distributed source. From the form of the distributed source terms, we find for species 1:

$$\frac{\partial s_1}{\partial u_1} = -C_{iv}u_2 - C_{sv}C_s \ ,$$

$$\frac{\partial s_1}{\partial u_2} = -C_{iv}u_1 \ ,$$

while for species 2, we find

$$\frac{\partial s_2}{\partial u_1} = -C_{iv}u_2 \ ,$$

$$\frac{\partial s_2}{\partial u_2} = -C_{iv}u_1 - C_{si}C_s \ .$$

BRHØ

This routine provides the boundary values for each species as given in Equation (7.6.2). The inner radius R_L is identified with side 1, and the outer radius is identified with side 3.

BRHØDT

This routine is not used since the boundary values do not involve time derivatives of the concentrations.

INDATA

This routine provides the initial data as given in Equation (7.6.3).

ANAL

This routine is not used in this problem and appears in dummy form.

The printed output is supplied on the following pages. Notice that output occurs after the first time step, the 100^{th} time step (since ISTDFQ=100), and the 141^{st} time step (at which the steady state solution has been found).

DI,DV 0.9377002431925776D-03 0.1983543028505083D-06

VEQL,XIEQL 0.1354356100587436D-06 0.6750441388724501D-17

VATV,CIV,CSV,CSI,CS
0.1658430801444832D-06 0.3406061271044640D+14 0.3707504460802126D+09 0.1752685867950059D+13 0.7387154739783821D-07

```
STORAGE MAXIMA FOR THIS COMPILATION :
MAXBRK          30
MAXSP            2
MAXTQD           4
MAXR             4
MXMENZ         100
MXNVAR         100
MAXGAP           2
MAXNOT          40
MXBGRD          20
MXZGBD          20
```

```
 READING NAMELIST  GRID


SPLINE ORDER IN R DIRECTION KR =  3
SPLINE ORDER IN Z DIRECTION KZ =  1
NUMBER OF SPECIES  NSPEC =   2
 LEFT BOUNDARY   RLOW =    0.1000000000000000D-05
 RIGHT BOUNDARY  RUP  =    0.1000000000000000D-04
 LOWER BOUNDARY  ZLOW =    0.0
 UPPER BOUNDARY  ZUP  =    0.1000000000000000D+01
GEOMETRY INDICATOR  DELTA =  2
NUMBER OF INTERFACES IN R AND Z  NTIR =  0 NTIZ =  0
NUMBER OF NON-INTERFACE MESH POINTS IN R AND Z   NMR = 10   NMZ =   1
 INITIAL CONTINUITY AT MESH POINTS   CONTR = 13           CONTZ = 13
NON-INTERFACE R MESH POINTS
I =  1 RMESH(I) =    0.1100000000000000D-05
I =  2 RMESH(I) =    0.1200000000000000D-05
I =  3 RMESH(I) =    0.1400000000000000D-05
I =  4 RMESH(I) =    0.2400000000000000D-05
I =  5 RMESH(I) =    0.4400000000000000D-05
I =  6 RMESH(I) =    0.6600000000000000D-05
I =  7 RMESH(I) =    0.8600000000000000D-05
I =  8 RMESH(I) =    0.9600000000000000D-05
I =  9 RMESH(I) =    0.9800000000000000D-05
I = 10 RMESH(I) =    0.9900000000000000D-05
NO NON-INTERFACE Z MESH POINTS
QUADRATURE ORDER FOR R DIRECTION  NQR =      3
QUADRATURE ORDER FOR Z DIRECTION NQZ =     1
CONSERVATIVE FORM INDEX FOR CONVECTION TERM CONSRV =  F
INDEX FOR ALGEBRAIC BOUNDARY CONDITIONS ALGBCS =  T
LOGICAL SWITCHES TO CONTROL PROGRAM
INDICATOR FOR STEADY STATE COMPUTATION STEDSW =  T
INDICATOR FOR TRANSIENT COMPUATION     TRANSW =  F
INDICATOR FOR INITIAL FIT OF DATA      INITSW =  T
INDICATOR FOR STEADY STATE RESTART     ISTDRS =  F
INDICATOR FOR TRANSIENT RESTART        ITRARS =  F
INDICATOR FOR READING STEADY STATE COEFFICIENTS FROM UNIT 10
                                       DUMPSW =  F
INDICATOR FOR ANALYTIC SOLUTION        IANAL  =  F
IREVLA(K) IS A SWITCH FOR INDICATING THAT THE COEFFICIENT OF THE TIME DERIVATIVE IS NOT IDENTICALLY ONE
IRHO(K)   IS A SWITCH FOR INDICATING THAT THE COEFFICIENT OF THE TIME DERIVATIVE IS IDENTICALLY ZERO
IREVLA( 1) =  F
IRHO( 1) =  F
IREVLA( 2) =  F
IRHO( 2) =  F
NUMBER OF USER SUPPLIED POINTS IN R DIRECTION  IRGRD =  4
RGRID( 1) =    0.3500000000000000D-05
RGRID( 2) =    0.5000000000000000D-05
RGRID( 3) =    0.8000000000000000D-05
RGRID( 4) =    0.9000000000000000D-05
NUMBER OF USER SUPPLIED POINTS IN Z DIRECTION  JZGRD =  1
ZGRID( 1) =    0.5000000000000000D+00


 READING NAMELIST  DATA


CONTR TOO HIGH, BEING RESET TO      3-1

CONTZ TOO HIGH, BEING RESET TO      1-1
```

```
NVGAP =  0     NHGAP =  0     LR = 11     LZ =  1     NR = 13     NZ =  1

II(I) =
 3   4   5   6   7   8   9  10  11  12  13

JI(J) =
 1

IREF(I) =
 1   1   1   1   1   1   1   1   1   1   1

JREF(J) =  1

HORIZONTAL ORDERING



NI=   2          NJ=  26          NCC= -28          MBW=   5



NIH=   1          NJH=  13          NCCH= -13          DM=   2
REQUIRED STORAGE FOR AL   NSTAL =      91
AVAILABLE STORAGE FOR AL    NAL =     100
REQUIRED STORAGE FOR PW   NSTPW =     416
AVAILABLE STORAGE FOR PW    NPW =     420


SYSTEM SIZE FOR THIS CASE
LR =    11     LZ =     1
NR =    13     NZ =     1
NVAR =   26
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  1
NEUMANN CONDITION FOR SPECIES  1 ON SIDE  2
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  3
NEUMANN CONDITION FOR SPECIES  1 ON SIDE  4
FOR SPECIES NO.  1
 SIDE  1 ALPHA =    0.10000000000000000D+01      BETA =    0.0                          GAMMA =    0.10000000000000000D+01
 SIDE  2 ALPHA =    0.0                          BETA =   -0.10000000000000000D+01      GAMMA =    0.10000000000000000D+01
 SIDE  3 ALPHA =    0.10000000000000000D+01      BETA =    0.0                          GAMMA =    0.10000000000000000D+01
 SIDE  4 ALPHA =    0.0                          BETA =    0.10000000000000000D+01      GAMMA =    0.10000000000000000D+01
DIRICHLET CONDITION FOR SPECIES  2 ON SIDE  1
NEUMANN CONDITION FOR SPECIES  2 ON SIDE  2
DIRICHLET CONDITION FOR SPECIES  2 ON SIDE  3
NEUMANN CONDITION FOR SPECIES  2 ON SIDE  4
FOR SPECIES NO.  2
 SIDE  1 ALPHA =    0.10000000000000000D+01      BETA =    0.0                          GAMMA =    0.10000000000000000D+01
 SIDE  2 ALPHA =    0.0                          BETA =   -0.10000000000000000D+01      GAMMA =    0.10000000000000000D+01
 SIDE  3 ALPHA =    0.10000000000000000D+01      BETA =    0.0                          GAMMA =    0.10000000000000000D+01
 SIDE  4 ALPHA =    0.0                          BETA =    0.10000000000000000D+01      GAMMA =    0.10000000000000000D+01

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  1     NS2=  0     NS3=  1     NS4=  0
 FOR SPECIES NO.  2     NS1=  1     NS2=  0     NS3=  1     NS4=  0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
FOR MATERIAL INDEX  1 AND SPECIES NO.  1 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
FOR MATERIAL INDEX  1 AND SPECIES NO.  2 HU1 =    0.10000000000000000D+01      HU3 =    0.10000000000000000D+01
```

```
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
FOR MATERIAL INDEX  1 AND SPECIES NO.  1 HU2 =     0.1000000000000000D+01      HU4 =     0.1000000000000000D+01
FOR MATERIAL INDEX  1 AND SPECIES NO.  2 HU2 =     0.1000000000000000D+01      HU4 =     0.1000000000000000D+01
ISTDFQ IS PRINTOUT FREQUENCY FOR STEADY STATE COMPUTATION

ISTDFQ =   100
TIME GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  NUTOUT =      3
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL  NUFREQ =  1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES  UTOUT =
   0.0                    0.1000000000000000D+01   0.2000000000000000D+01
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  T
NUMBER FOR GRAPHS ASSOCIATED WITH THIS RUN
      1
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
LOCAL TEMPORAL ERROR CONTROL  EPS =    0.1000000000000000D-02
 INITIAL TIME STEP           HINIT =    0.1000000000000000D-14
 MAXIMUM ORDER OF TIME INTEGRATION
                            MXGORD =  5
INITIAL TIME  T0 =    0.0
CONTINUITY FOR R AND Z DIRECTIONS MAY HAVE BEEN RESET
CONTR =  2          CONTZ =  0
DEFAULT INITIAL COEFFICIENTS   R =


   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01   0.1000000000000000D+01
   0.1000000000000000D+01   0.1000000000000000D+01
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1354337509129275D-06   0.1354315433884149D-06   0.1354358135060917D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  1,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724501D-17   0.6750441388724502D-17   0.6750441388724502D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  2,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1354406832282897D-06   0.1354440384084481D-06   0.1354397590904382D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  2,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724500D-17   0.6750441388724499D-17   0.6750441388724499D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  3,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1354315400354157D-06   0.1354195965397319D-06   0.1354186427943693D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  3,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724500D-17   0.6750441388724502D-17   0.6750441388724502D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  4,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1354307335398893D-06   0.1354515657878788D-06   0.1354498148398527D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  4,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724501D-17   0.6750441388724499D-17   0.6750441388724500D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  5,  1)-TH RECTANGLE.
```

```
(FROM INIFIT)
   0.1354348571594631D-06   0.1354184156490662D-06   0.1354301290179933D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  5,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724501D-17   0.6750441388724502D-17   0.6750441388724502D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  6,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1354484598565673D-06   0.1354575866771685D-06   0.1354302741632372D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  6,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724501D-17   0.6750441388724500D-17   0.6750441388724501D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  7,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1354036513453517D-06   0.1354090358587570D-06   0.1354857446171248D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  7,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724501D-17   0.6750441388724502D-17   0.6750441388724501D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  8,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1355325791005313D-06   0.1354449361062371D-06   0.1351623411566703D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  8,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724500D-17   0.6750441388724500D-17   0.6750441388724499D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE (  9,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1350502403639819D-06   0.1355639529963101D-06   0.1368375607278773D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE (  9,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724499D-17   0.6750441388724499D-17   0.6750441388724501D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE ( 10,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1375079581940802D-06   0.1362835163270871D-06   0.1323260816574370D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE ( 10,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724502D-17   0.6750441388724502D-17   0.6750441388724501D-17
 VALUES OF CONCENTRATION U OF THE  1 TH SPECIES AT THE QUADRATURE POINTS IN THE ( 11,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.1295280297077747D-06   0.1355066841257855D-06   0.1567798469216146D-06
 VALUES OF CONCENTRATION U OF THE  2 TH SPECIES AT THE QUADRATURE POINTS IN THE ( 11,  1)-TH RECTANGLE.
(FROM INIFIT)
   0.6750441388724500D-17   0.6750441388724500D-17   0.6750441388724500D-17


FROUT FOR TIME =   -0.9999999999999d99D-01

  W =
   0.1354356100587436D-06   0.6750441388724491D-17   0.1354281452254982D-06   0.8132980341394767D-17   0.1354530400485813+D-06
   0.7668247758590022D-17   0.1354104384229703D-06   0.7759809512440088D-17   0.1354703851088253D-06   0.7747884397333644D-17
   0.1353944199291734D-06   0.7752136092111383D-17   0.1354879527752190D-06   0.7748617246103605D-17   0.1353495990136995D-06
   0.7753311219555073D-17   0.1356073960204664D-06   0.7744777247814656D-17   0.1349307072280626D-06   0.7767034183430563D-17
   0.1385610094087981D-06   0.7647794830658687D-17   0.1227612639323376D-06   0.8166725114482842D-17   0.1658430801444832D-06
   0.6750441388724501D-17

 VALUES OF CONCENTRATIONS U FOR THE   1 TH SPECIES ON THE USERS GRID
(FROM FROUT VIA STEADY)

APPROXIMATION IS
   0.1354181484122020D-06   0.1354566485068181D-06   0.1354397421494989D-06   0.1354862352073338D-06

 VALUES OF CONCENTRATIONS U FOR THE   2 TH SPECIES ON THE USERS GRID
(FROM FROUT VIA STEADY)
```

```
APPROXIMATION IS
   0.7751055231927171D-17   0.7749775046560578D-17   0.7750322285162425D-17   0.7748768915838959D-17
 AT TOUT =    -0.9999999999999899D-01 SOBN =     0.3454704102123250D+01 AND HUSED =     0.1000000000000000D-14
NQUSED =  1

SUPN =
   0.3454704102123250D-14
  1 CALLS TO DRIVE HAVE OCCURRED
A STEADY PRINT TAKES LESS THAN A CENTISEC.--USE 1 CENTISEC. AS AN ESTIMATE

         A CALL TO DRIVE TAKES LESS THAN A CENTISEC.-- USE 1 CENTISEC. AS A ESTIMATE


PROUT FOR TIME =    -0.9999949717053860D-01

  W =
   0.1354356100587436D-06   0.6750441388724359D-17   0.1355659494385751D-06   0.9199765552810824D-12   0.1357316412114062D-06
   0.2418388107133931D-11   0.1359031915222312D-06   0.4350126792762649D-11   0.1356034387588790D-06   0.9166474202328652D-11
   0.1363846926207798D-06   0.1198910195857619D-10   0.1352622148647580D-06   0.1066976884983774D-10   0.1370247027433115D-06
   0.6857184826960862D-11   0.1337617468416721D-06   0.2823937842640330D-11   0.1511128768348078D-06   0.9635191516182834D-12
   0.1584041348015896D-06   0.4858689642827705D-12   0.1633337051542032D-06   0.1633412265383144D-12   0.1658430801444832D-06
   0.6750441388724501D-17

 VALUES OF CONCENTRATIONS U FOR THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA STEADY)

APPROXIMATION IS
   0.1361175331012343D-06   0.1356418727318245D-06   0.1358832686206475D-06   0.1364667855489536D-06

 VALUES OF CONCENTRATIONS U FOR THE   2 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA STEADY)

APPROXIMATION IS
   0.1141800004184383D-10   0.1088675613448436D-10   0.5703053936872782D-11   0.3059871655289187D-11
 AT TOUT =    -0.9999949717053860D-01 SOBN =     0.7855616739644008D+03 AND HUSED =     0.1287566125049381D-06
NQUSED =  2

SUPN =
   0.2326507453637270D-08   0.6511623408241306D-09
100 CALLS TO DRIVE HAVE OCCURRED


PROUT FOR TIME =    -0.9945437428639980D-01

  W =
   0.1354356100587436D-06   0.6750441388724534D-17   0.1413857385001132D-06   0.9007178006423006D-12   0.1510942745432073D-06
   0.2367541285396863D-11   0.1636334946404234D-06   0.4258351909129336D-11   0.1951304531538425D-06   0.8969606412150964D-11
   0.2154707528866665D-06   0.1172388584941036D-10   0.2126962311270903D-06   0.1045236793664319D-10   0.1967350570457871D-06
   0.6745131053757933D-11   0.1786903806731880D-06   0.2786362132784717D-11   0.1702366639047008D-06   0.9514190672420399D-12
   0.1680603280957358D-06   0.4799000956306796D-12   0.1665890491054867D-06   0.1613780644175671D-12   0.1658430801444832D-06
   0.6750441388724501D-17

 VALUES OF CONCENTRATIONS U FOR THE   1 TH SPECIES ON THE USERS GRID
(FROM PROUT VIA STEADY)

APPROXIMATION IS
   0.2123251491692725D-06   0.2128430658711645D-06   0.1915245130818414D-06   0.1797285796021015D-06

 VALUES OF CONCENTRATIONS U FOR THE   2 TH SPECIES ON THE USERS GRID
```

```
(FROM PROUT VIA STEADY)

APPROXIMATION IS
   0.1116889899746983D-10   0.1066021215513294D-10   0.5610814796935097D-11   0.3016755327895812D-11
 AT TOUT =   -0.9945437428639980D-01 SOBN =    0.4746234182444423D-07 AND HUSED =    0.1650715154623751D-03
NQUSED =  1

SUPN =
   0.7834680692354277D-11
141 CALLS TO DRIVE HAVE OCCURRED
 STEADY-STATE SOLUTION HAS BEEN ACHIEVED
COEFFICIENTS WRITTEN ON DUMP DATASET


 READING NAMELIST  DATA
```

```
 CHANGES IN NAMELIST DATA MAY HAVE BEEN MADE FOR TRANSIENT
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  1
NEUMANN CONDITION FOR SPECIES  1 ON SIDE  2
DIRICHLET CONDITION FOR SPECIES  1 ON SIDE  3
NEUMANN CONDITION FOR SPECIES  1 ON SIDE  4
FOR SPECIES NO.  1
 SIDE  1 ALPHA =     0.1000000000000000D+01      BETA =     0.0                          GAMMA =     0.1000000000000000D+01
 SIDE  2 ALPHA =     0.0                         BETA =     0.1000000000000000D+01       GAMMA =     0.1000000000000000D+01
 SIDE  3 ALPHA =     0.1000000000000000D+01      BETA =     0.0                          GAMMA =     0.1000000000000000D+01
 SIDE  4 ALPHA =     0.0                         BETA =     0.1000000000000000D+01       GAMMA =     0.1000000000000000D+01
DIRICHLET CONDITION FOR SPECIES  2 ON SIDE  1
NEUMANN CONDITION FOR SPECIES  2 ON SIDE  2
DIRICHLET CONDITION FOR SPECIES  2 ON SIDE  3
NEUMANN CONDITION FOR SPECIES  2 ON SIDE  4
FOR SPECIES NO.  2
 SIDE  1 ALPHA =     0.1000000000000000D+01      BETA =     0.0                          GAMMA =     0.1000000000000000D+01
 SIDE  2 ALPHA =     0.0                         BETA =     0.1000000000000000D+01       GAMMA =     0.1000000000000000D+01
 SIDE  3 ALPHA =     0.1000000000000000D+01      BETA =     0.0                          GAMMA =     0.1000000000000000D+01
 SIDE  4 ALPHA =     0.0                         BETA =     0.1000000000000000D+01       GAMMA =     0.1000000000000000D+01

 SIDE INDICATORS BY SPECIES

 FOR SPECIES NO.  1     NS1=  1     NS2=  0     NS3=  1     NS4=  0
 FOR SPECIES NO.  2     NS1=  1     NS2=  0     NS3=  1     NS4=  0
BOUNDARY H FUNCTION FOR SIDES 1 AND 3
FOR MATERIAL INDEX  1 AND SPECIES NO.  1 HU1 =     0.1000000000000000D+01      HU3 =     0.1000000000000000D+01
FOR MATERIAL INDEX  1 AND SPECIES NO.  2 HU1 =     0.1000000000000000D+01      HU3 =     0.1000000000000000D+01
BOUNDARY H FUNCTION FOR SIDES 2 AND 4
FOR MATERIAL INDEX  1 AND SPECIES NO.  1 HU2 =     0.1000000000000000D+01      HU4 =     0.1000000000000000D+01
FOR MATERIAL INDEX  1 AND SPECIES NO.  2 HU2 =     0.1000000000000000D+01      HU4 =     0.1000000000000000D+01
ISTDFQ IS PRINTOUT FREQUENCY FOR STEADY STATE COMPUTATION

ISTDFQ =   100
TIME GRID FOR PROUT

NUMBER OF MAJOR TIME VALUES  NUTOUT =     3
NUMBER OF SUBINTERVALS OF EACH MAJOR TIME INTERVAL  NUFREQ =  1
(OUTPUT WILL OCCUR AT EACH SUCH TIME)
MAJOR TIME VALUES  UTOUT =
   0.0                      0.1000000000000000D+01   0.2000000000000000D+01
DATASET CREATED FOR USE IN GRAPHICS
 GRAPH =  T
NUMBER FOR GRAPHS ASSOCIATED WITH THIS RUN
     1
PRINT SWITCH INDICATORS
IPRSW1 = 0      IPRSW2 = 0      IPRSW3 = 0      IPRSW4 = 0      IPRSW5 = 0
ODE PACKAGE DATA
LOCAL TEMPORAL ERROR CONTROL  EPS =    0.1000000000000000D-02
 INITIAL TIME STEP           HINIT =    0.1000000000000000D-14
 MAXIMUM ORDER OF TIME INTEGRATION
                            MXORD =  5
INITIAL TIME  T0 =    0.0
CONTINUITY FOR R AND Z DIRECTIONS
CONTR =  2          CONTZ =  0
INITIAL COEFFICIENTS FOR TRANSIENT   W =


   0.1354356105587436D-06   0.6750441388724534D-17   0.1413857386500115[illegible]D-06   0.9037178006423006D-12
   0.1510942745432073D-06   0.2367541285396863D-11   0.1636334946404234D-06   0.4258351939129836D-11
   0.1951304531538425D-06   0.8969606412150964D-11   0.2154707528866665D-06   0.1172388594941036D-10
   0.2126962311270903D-06   0.1045236793664319D-10   0.1967350570457871D-06   0.6745131053757933D-11
   0.1786953606731880D-06   0.2786362132784717D-11   0.1702366639047008D-06   0.9514190672420399D-12
```

0.1680603280957358D-06 0.4799000956306796D-12 0.1665890491054867D-06 0.1613780644175671D-12
0.1658430801444832D-06 0.6750441388724501D-17

END OF CASE

We next consider the use of the Cross Section Plotting (CSP) graphics program in connection with the defect-diffusion problem.

The CSP program is provided with a dummy form of SUBROUTINE ANAL which is the only user-supplied routine. Since we do not have an analytic solution, we leave the dummy routine as it is.

There are two Namelists required: Namelist FORMAT and Namelist CSPIN.

Namelist FORMAT

1. Iterative or direct indicator

   The direct version is used.

   ITRTV=0, (Default)

2. Number of curves to be produced per time value

   This variable specifies the number of curves to be produced for each time value. Since there are two equations and each solution is produced once for each time value, use

   IGNUM=2,

3. Number of grid points for graphical purposes

   This number cannot exceed the macro variable NRES1 which has a value of 501.

   NRESIN=NRES1, (Default)

4. Logical variable for doing space-time plots

   A time plot is not called for.

   ITIME=F, (Default)

Namelist CSPIN

1. Indicator for grouping format

   The IFØRMT, ISPEC, LGRØUP, and LØRDER variables describe how the curves are to be displayed. For each time value in this example, a single frame will be produced. Within this frame both curves will be plotted and, since these solutions differ considerably in magnitude, they will be put on separate sets of axes. Thus the separate mode is used.

   IFØRMT=0, (Default)

2. Estimate of the minimum value of the ordinates

   This is defaulted to zero.

   YAXMIN=0.0, (Default)

3. Estimate of the maximum value of the ordinate for all curves.

   This defaults to one. The CSP program will automatically rescale this value since it is much larger than necessary.

   YAXMAX=1.0, (Default)

4. Species number for each curve

   The next three variables will be two component vectors since IGNUM=2. Associate the first curve with the first specie and the second curve with the second specie. (See Note 10 of the Machine Readable Documentation for more general examples of ISPEC, LGRØUP, and LØRDER.)

   ISPEC=1,2,

5. Frame number indicator

   Since only one frame is generated for each time value:

   LGRØUP=1,1, (Default)

6. Ordering of curves on each frame

   The plots will be ordered from the bottom to the top of each frame.

   LØRDER=1,2,

7. R coordinate of first endpoint of cross-section

   This is a one-dimensional problem in the R direction. The cross-section will be from $(10^{-6},0)$ to $(10^{-5},0)$.

   A1=1.E-6,

8. R coordinate of second endpoint of cross-section

   A2=1.E-5,

The CSP program will provide a frame for each time that printout occurred. Thus there will be a frame corresponding to the 1^{st}, 100^{th}, and 141^{st} time step. The following page provides the graph associated with the 141^{st} step (the steady-state solution). Notice that the time value appearing in the graph is artificial and is only used within DISPL to control integration to steady-state.

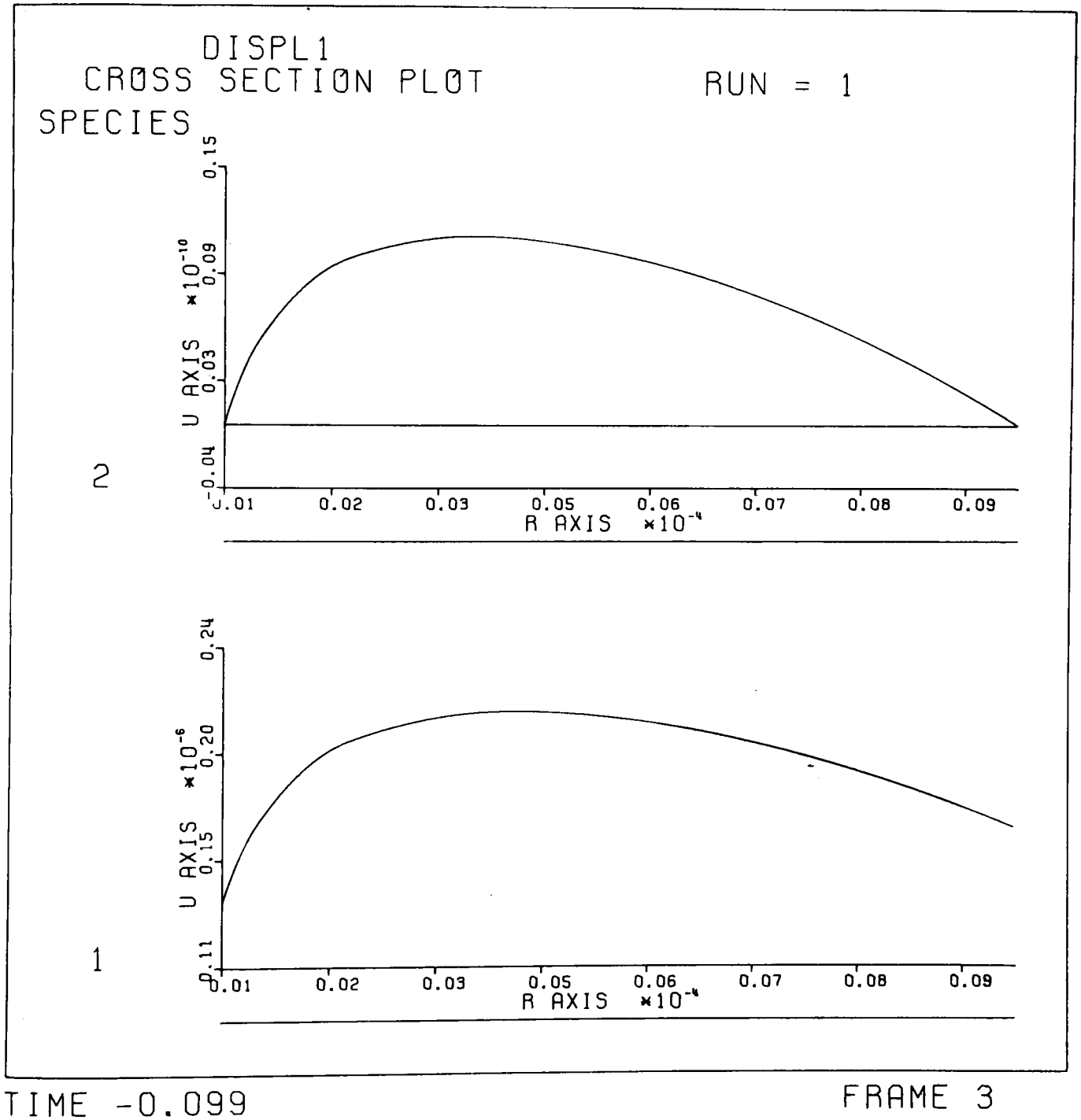


Figure 7.6.1

Graph of Concentrations at 141st Time Step

266

APPENDIX A

Machine Readable Documentation

DISPL1: ONE AND TWO DIMENSIONAL KINETICS-DIFFUSION SOFTWARE

This is documentation for DISPL1, a one and two dimensional kinetics-diffusional software package. This package is described in DISPL: A SOFTWARE PACKAGE FOR ONE AND TWO SPATIALLY DIMENSIONED KINETICS-DIFFUSION PROBLEMS by G.K. Leaf, M. Minkoff, G.D. Byrne, D. Sorensen, T. Bleakney, and J. Saltzman, Argonne National Laboratory Applied Mathematics Division Report ANL-77-12, May, 1977.

Table of Contents

1. Role

There is one read of Namelist GRID followed by two reads of Namelist DATA for each complete job case. GRID contains input variables that determine the structure of the case: positions of interfaces, selection of steady-state solution, transient solution, etc.

Namelist DATA for the current case is read once immediately after GRID and before any calls to the calculation routines. It is read again immediately before the call to TIMEX, the transient routine. This Namelist contains boundary values, gap data, reaction coefficients, output switches, and GEAR control variables which can be modified via the second read. If no changes are required, only a null DATA Namelist need be supplied for the second read. Each call to EXEC (the highest level routine in DISPL1) requires the presence of these 3 Namelist sets, but any or all may be null.

2. Format of namelist input

Namelist uses a free-format similiar to ordinary assignment statements, as described in the IBM Fortran IV Ref. manual, pages 54,55. Only the namelist identifier, either GRID or DATA , is fixed in its format. Beginning in column 2, enter

```
    &GRID free-format assignments, . . .
       . . . &END
```

This is followed by:

```
    &DATA . . . &END
```

followed by the second read of DATA :

```
 &DATA . . .  &END
```

where again the string &DATA begins in column 2.

A null entry for either namelist is of the form:

```
    &GRID &END   or:
    &DATA &END
```

Note that the input assignments within each namelist can be supplied in any order. Any variable can be omitted entirely if its default value is desired.

As explained in the reference cited above, array data can be entered either with or without subscripts. Examples of each are:

```
    MATL(1,1)=1,MATL(1,2)=3,
    MATL(2,1)=4,MATL(2,2)=2,

    IFTYPR=1,0,1,1,IFTYPZ=0,0,
```

It is recommended, as shown above, that 2 and 3 dimensional data be entered with explicit subscripts for readability. One-dimensional data can be entered more conviently without subscripts.

The description of Namelist for CDC software is given in in the CDC Fortran Extended Version 4 Reference Manual, pp. I-9-15, I-9-17. The above discussion applies for CDC Namelist with & replaced by $.

3. Variables in Namelist GRID

The list here shows the default values if the variable is omitted. Dimensioning is shown in parentheses.

```
Spline order
    KR=4,KZ=4,
Quadrature order
    NQR=MAX(KR,1),NQZ=MAX(KZ,1),      (The default is set using the
                                       values of KR and KZ)
Number of species
    NSPEC=1,
Geometry indicator: 0=rectangular, 1=cylindrical, (2=spherical,
                                                    1D only)
    DELTA=0,
Domain boundaries
    RLOW=0.D0,RUP=1.D0,ZLOW=0.D0,ZUP=1.D0,
Number of interfaces
    NTIR=0,NTIZ=0,
Interface mesh points
    RIF=0.D0,0.D0, . . ,ZIF=0.D0, . . .,
Interface type codes: 0=gap, 1=continuous
    IFTYPR=1,1, . . .,IFTYPZ=1,1, . . .,
Number of noninterface mesh points (in R or Z direction)
  in the domain (not including the end points). If the RMESH (ZMESH)
  vector is omitted, the code will generate NMR (NMZ) equally
  space interior points.
    NMR=0,NMZ=0,
```

```
Non-interface mesh points
  If one of these vectors is omitted, the code generates equally
  spaced interior points in that coordinate direction.
    RMESH=0.D0, ..,ZMESH=0.D0, . . .,
Continuity of mesh points
    CONTR=KR-1,CONTZ=KZ-1, (Defaults set internally)
Continuity index at noninterface mesh points in R direction
    INUR(I),(I=1,NMR)      (Defaults to CONTR)
Continuity index at noninterface mesh points in Z direction
    INUZ(J),(J=1,NMZ)      (Defaults to CONTZ)
Material table   (See Note 1 below)
    MATL=1,1, . . .,  (MAXBRK,MAXBRK)
Index for algebraic boundary conditions
    ALGBCS=T,  (T means that algebraic boundary conditions
                are used as provided by BRHO.  In this case BRHODT
                can be a dummy routine.  F means that BRHODT will be
                used.  BRHODT must return the time derivative of the
                essential boundary conditions; but NOT the time
                derivative of the non-essential conditions.  In this
                case BRHO may be a dummy routine).
Conservative form index for convection term
    CONSRV=T,  (T means conservative form is used,
                F means substantial derivative form is used)
Switch for indicating the presence of heat capacity coefficient
  in the partial time derivative term.  For the K-th species,
  set IREVLA(K)=F, if RHOCP(K,T,R,Z,U) is identially 1.0.
  Otherwise set IREVLA(K)=T.
    IREVLA(K)=F,    (K=1,NSPEC)
Indicator for RHOCP(K) being identically zero.  IRHO(K)=T,
  implies that RHOCP is identically zero.  Note that IREVLA(K)=F and
  IRHO(K)=T is inconsistent.  If the user does this, the code will
  issue a warning and reset IREVLA(K) to .TRUE.
    IRHO(K)=F,
Selection of calls (in order) to
  INITIT,STEADY,TIMEX:
    INITSW=T,STEDSW=T,TRANSW=T,   (Default values)
Frequency of output from STEADY
    ISTDFQ=100,     (Output from steady occurs every ISTDFQ time steps
                     except that the first output occurs after the first
                     time step.)
                     Default value is 100. NEVER USE ISTDFQ=0!
 Number of points in the R direction for user's output grid.
   This grid is optional and provides an alternative to giving
   the approximate and analytic solution at the Gaussian points.
   The alternative grid is the tensor product of RGRID and ZGRID.
   IRGRD=0 implies that the output is given at the Gaussian
   points.  IRGRD=1 should be used when the problem is one dimensional
   in the Z direction.
     IRGRD=1,
 Points in the R direction for user's grid
   If this vector is omitted and if IRGRD > 2 then the code generates
   equally spaced points including the end points.
     RGRID(I),  (I=1,...IRGRD)   (RGRID(I)=0.5D0)
 Number of points in the Z direction for user's output grid.  JZGRD=0
   implies that the output is given the the Gaussian points.  JZGRD=1
   should be used when the problem is one dimensional in the R
   direction.
     JZGRD=1,
 Points in the Z direction for user's grid
```

If this vector is omitted and if JZGRD > 2 then the code generates equally spaced points including the end points.
ZGRID(J), (J=1,...JZGRD) (ZGRID(J)=0.5D0)

Indicator for giving the analytic solution on the grid (either the Gaussian or user grid). The analytic solution is computed in SUBROUTINE ANAL. (See Note 8).
IANAL=F

Indicator for restarting STEADY
This indicator and the next two indicators can be used to restart a calculation from a dump generated during a previous run. The code dumps information under the following four conditions: if it detects that time is expiring during steady-state calculation; at the normal conclusion of a steady-state calculation; if it detects that time is expiring during a transient calculation; and at the normal conclusion of a transient calculation. Only the last dump during a given run is meaningful since each dump writes from the beginning of the file. Thus to generate a dump at the normal conclusion of the steady-state calculation set TRANSW=F. The ISTDRS indicator is associated with restarting the first type of dump; the DUMPSW indicator is associated with restarting the second type; and the ITRARS indicator is associated with the third type. The fourth type of dump can be utilized as follows: Set up an array of UTOUT of dimension N2 and set NUTOUT=N1<N2. The program will then dump at T=UTOUT(N1). On the restart run set INITSW, GUESSW and STEDSW to false and TRANSW to true and use NUTOUT=N2. The calculation will restart from UTOUT(N1) and proceed to UTOUT(N2). To restart a steady-state calculation set ISTDRS=T, INITSW=F, GUESSW=F, and STEDSW=T. The code will then read the current value of the approximating coefficients and time (variables W and T, respectively from unit 10.
ISTDRS=F,

Indicator for reading coefficients from unit 10.
If the dump in a previous run occurred at the normal conclusion of the steady-state calculation, the code dumped the final coefficients. By setting DUMPSW=T, these coefficients can be read from unit 10 and can be used in either a new steady-state or transient calculation.
DUMPSW=F,

Indicator for restarting a transient calculation.
If ITRARS=T, then the current value of the approximating coefficients and time (variables W and T, respectively) are read from unit 10. When ITRARS is true, INTISW, GUESSW, and STEDSW must be false and TRANSW must be true.
ITRARS=F,

Indicator for order of derivatives computed on the user supplied output grid. The code always provides the approximate solution on the grid (either user supplied or Gaussian). However, if a user supplied grid is used, partial derivatives of the approximation can be requested. IORD is a vector of, at most, fifteen components. Each component is an integer of the form 10*I+J and will cause the mixed partial of order I with respect to R and order J with respect to Z to be computed. Thus IORD=10,11, will cause DU(K)/DR and DU(K)/(DR DZ) to be computed as well as U(K) for all K. Notice that the user need not specify the I=J=0 case (function value case).
IORD=0,0,...,
&END

4. Variables in Namelist DATA

```
Spline coefficients
    W=1.D0, . . .,          (MXNVAR)
Boundary value coefficients  (MAXSP,4)  (See Note 5)
    ALPHA=0.D0, . .,BETA=1.D0, . .,GAMMA=0.D0, . .,
Boundary condition switches
  NSI(K) is the switch for species K on side I.
  NSI(K)=0  means BETA is non-zero for species K on side I.
  NSI(K)=1  means BETA is zero for species K on side I.
  NSI(K)=-1 means that no boundary condition is imposed for
            species K on side I.
  Note that NSI(K) can vary with K.
  When specifying the boundary condition switches, the user
  must be sure to also specify consistent values of ALPHA(K,I),
  BETA(K,I), and GAMMA(K,I).  Notice that when the logical
  indicators DRCHLT(K,I) or NEUMAN(K,I) are used, the appropriate
  boundary condition switches and coefficients are set automatically.
    NS1(K),NS2(K),NS3(K),NS4(K)=0,       K=1,NSPEC
Dirichlet boundary condition indicator (MAXSP,4)
  If the K-th species on the I-th side has a Dirichlet boundary
  condition (U(K) specified on side(I)), then setting DRCHLT(K,I)=T,
  will set NSI(K)=1, ALPHA(K,I)=1.D0, BETA(K,I)=0.D0, and
  GAMMA(K,I)=1.D0.  Thus for a Dirichlet condition one can use
  DRCHLT(K,I)=T, in place of setting NSI(K), ALPHA(K,I), BETA(K,I),
  and GAMMA(K,I).
    DRCHLT(K,I)=F,   K=1,MAXSP     I=1,4
Neumann boundary condition indicator   (MAXSP,4)
  If the K-th species on the I-th side has a Neumann boundary
  condition (the normal derivative of U(K) specified on side I),
  then setting NEUMAN(K,I)=T, will set NSI(K)=0, ALPHA(K,I)=0.D0,
  BETA(K,I)=1.D0, and GAMMA(K,I)=1.D0.  The negative sign
  associated with the normal derivative on sides 1 and 2 is taken
  care of in the program.  Thus for a Neumann condition, one can use
  NEUMAN(K,I)=T, in place of setting NSI(K), ALPHA(K,I), BETA(K,I),
  and GAMMA(K,I).  Note that the symmetric condition (homogeneous
  Neumann condition) is a default in the program so that in this
  case one does not have to set any boundary condition indicator.
    NEUMAN(K,I)=F,    K=1,MAXSP    I=1,4
Boundary h function for each side  (MAXBRK,MAXSP) (See Note 6)
    HU1=1.DC, . .,HU2=1.D0, . .,
    HU3=1.D0, . .,HU4=1.D0, . .,
Gap coefficients      (MAXBRK,MAXBRK,MAXSP)  (See Note 2)
    HVGAP=0.D0, . .,HHGAP=0.D0, . .,
Reaction coefficients, first order     (See Note 3)
    CK=0.D0, . .,     (MAXSP,MAXSP)
Reaction coefficients, second order    (See Note 4)
    CKK=0.D0, . .,     (MAXSP,MAXSP,MAXSP)
Initial time for start of transient calculation
    T0=0.D0,
Output time control for TIMEX through PROUT
    NUTOUT=3,NUFREQ=1,UTOUT=0.D0,1.D0,2.D0, 97*0.D0,  (See Note 7)
Graphics output switch.  Output for later graphics analysis
  will be written on unit 12 if this switch is true.
    GRAPH=F,
Number of run.  This integer variable is used in the graphics
  programs to identify the run which produced the graphics file
  on unit 12.
    NUMGRF=1,
ODE solver control parameters
```

```
    EPS=1.D-4,HINIT=1.D-3,
Maximum order used in ODE solver
    MXGORD=5,       (MXGORD must be between 1 and 5)
Internal output control switches.  If equal to 1 the print
  will occur.  If equal to 0 the print will not occur.
  IPRSW1 controls certain prints in INPROC
  IPRSW2 controls prints in INIFIT and certain prints in INPROC
  IPRSW3 controls prints in GFUN and PEDERV
  IPRSW4 controls prints in GUESS1, RHS, TIMEX and BDALHS
  IPRSW5 controls prints in PEDERV
    IPRSW1=0,IPRSW2=0,IPRSW3=0,IPRSW4=0,IPRSW5=0,
    &END
```

5. Graphics namelists

These namelists are used in one or more of the three graphics packages--CSP (cross section plots), CONTOR (contour plots), and THREED (three dimensional plots via the DISPLAA graphics software). Namelist FORMAT is used in all three programs. Namelist CSPIN is used only in CSP. Namelist CNTRIN is used only in CONTOR while namelist DIM3IN is used only in THREED. All variables in the graphics namelists are in SINGLE PRECISION.

a. Variables in Namelist FORMAT

This namelist provides basic data dealing with the ordering of the coefficients of the approximating and the number of curves to be plotted.

```
Indicator for iterative or direct version of the code.  This
  indicator estalishes the ordering assumed of the coefficients.
  If ITRTV is set to 1, the ordering is that of the iterative
  version.  If ITRTV is anything else, the ordering is assumed
  to be that of the direct version.  (The Argonne Code Center
  version is the direct version).
    ITRTV=0,
Number of curves (or contours or three dimensional views) to
  be produced at each time value.  Usually this is
  the number of species, NSPEC.
    IGNUM=1,
Number of graphics grid points in each coordinate direction.
  For CSP this value defaults to NRES1.  For CONTOR and THREED
  this value defaults to NRESD.
    NRESIN=NRES1 or NRESD,   (That is, NRESIN is set to the value
                              of the Macro variable NRES1 or NRESD).
Logical variable for doing space-time plots,
  If ITIME=T, the contour plot uses the horizontal axis for time.
  In the case of CONTOR and THREED the vertical axis is used for the
  spatial direction.
    ITIME=F,
Integer variable for the number of time values used in space-time
  plots.  If NTIME is set less than the actual number of
  time values, only the first NTIME values will be used.  If
  NTIME is set greater than or equal to the actual number of
  time values, all of the values will be used, although a warning
  will be printed if NTIME is greater than the actual value.
    NTIME=NRES1, for CSP   (That is, NTIME defaults to the macros NRES1
                            in the CSP program).
```

```
    NTIME=NRESD, for CONTOR and THREED
    &END
```

b. Variables in Namelist CSPIN

This namelist provides data dealing specifically with the CSP program.

```
Indicator for cinema mode.  If ICN is 1 the code uses cinema
  mode to produce multiple copies of a frame.  If ICN is anything
  else, cinema mode is not used and only one copy of a frame may
  be requested.
    ICN=0,
Indicator for grouping format of frames.  If IFORMT is set to 1
  the frames will be produced in packed format (more than one curve
  on each set of axes).  If IFORMT is anything else the separate
  mode will be used.  That is, we would produce multiple sets
  of axes within each frame.
    IFORMT=0,
Logical indicator for the analytic solution.  If IANAL is set TRUE,
  the analytic solution is graphed along with the approximate
  solution.  (See Note 9).
    IANAL=F,
Estimate of the minimum value of the ordinate for all curves in a
  given frame.  This estimate is checked by the code and is used if
  YAXMIN and YAXMAX are reasonably close to the actual values used.
  (These estimates can be used to obtain a constant vertical axis
  range for all plots, e.g. for use in movie generation).  A separate
  estimate is made for each frame defined in LGROUP.  Thus YAXMIN
  and YAXMAX have max(LGROUP) components.
    YAXMIN=0.0,0.0,
Estimate of the maximum value of the ordinate for all curves in a
  given frame.
    YAXMAX=1.0,1.0,
The species number being referred to for each curve.  (Recall that
  there are IGNUM curves for each time value.  Thus ISPEC, LGROUP,
  and IORDER are vectors of size IGNUM).
    ISPEC=1,2,3,4,...,MXGNUM,
The frame in which this curve will be plotted.  That is, we can
  make several frames for each time value and we can have different
  combinations of curves on each frame.  (See Note 10).
    LGROUP=1,1,...,
The order in which the plots will appear on a given frame (in separate
  mode) or the calcomp plotting symbol used (in packed mode) is
  indicated by the integer variable LORDER.  The plots will be ordered
  from the bottom to the top of the frame (if the separate mode is
  used).  If the packed mode is used, this variable will only effect
  the choice of the calcomp plotting symbol associated with each
  curve.  The Ith curve will be associated with the LORDER(I)th calcomp
  plotting symbol.  The first two calcomp symbols are a
  square and an octagon.
    LORDER=1,1,...,
R coordinate of first endpoint of the cross-section.  (This and the
  following three variables define the endpoints of the straight line
  cross-section to be used for each frame.  Since there can be more
  than one frame for each time value, these variables are vectors of
  dimension max(LGROUP).  That is, the points (A1(M),B1(M)) and
  (A2(M),B2(M)) define the cross-section for the Mth frame).
    A1=0.0,0.0,...,
```

```
Z coordinate of first endpoint of the cross-section.
    B1=0.0,0.0,...,
R coordinate of other endpoint of the cross-section.
    A2=1.0,1.0,...,
Z coordinate of other endpoint of the cross-section.
    B2=1.0,1.0,...,
    &END
```

c. Variables in Namelist CNTRIN

This namelist provides the dimensions of the R-Z rectangle for each species to be used in contour or three-dimensional plotting. It is possible to use a different rectangle for each species. Also variables ITIME and NTIME are given here for use in plotting solution contours for one spatial dimension problems over space and time. This namelist is read only once and provides species numbers and the R-Z coordinates as vectors over the species index. It should be noted that the CONTOR program does not actually compute the contours. Rather, it processes the user's input and writes out a formatted dataset on unit 13. This dataset consists, for each contour plot, of a series of card images. All of these images, except the last, are values of R, Z, and function value for a point in the contouring grid. These images are written with a 1X,3E14.6 format. The last card for each contour plot is written with a 43X,'END' format. While this formatted dataset is suitable for use at Argonne with the in-house contouring package CONTOUR.BLACKBOX, the information on the dataset can be used to interface to other contouring packages. The CONTOUR.BLACKBOX package is not available for distribution.

```
Integer vector of species numbers. This vector provides the species
  number to be plotted in the Mth frame.
    ISPEC=1,2,3,4,...,MAXSP,
Real vector of minimum R coordinate to be plotted for the Mth frame,
  i.e. RMIN(I).
    RMIN=0.0,0.0,...,
Real vector of maximum R coordinate to be plotted for the Mth frame,
  i.e. RMAX(I).
    RMAX=1.0,1.0,...,
Real vector of minimum Z coordinate to be plotted for the Mth frame,
  i.e. ZMIN(I).
    ZMIN=0.0,0.0,...,
Real vector of maximum Z coordinate to be plotted for the Mth frame,
  i.e. ZMAX(I).
    ZMAX=1.0,1.0,...,
Integer variable for the number of grid points used for the
  R coordinate direction in contour plotting.
    NRESR=NRESIN,     (This variable defaults to the value given
                       to NRESIN in Namelist FORMAT)
Integer variable for the number of grid points used for the
  Z coordinate direction in contour plotting.
    NRESZ=NRESIN,     (This variable defaults to the value given
                       to NRESIN in Namelist FORMAT)
    &END
```

d. Variables in Namelist DIM3IN

This namelist provides a viewpoint for the three-dimensional plot and bounds the function axis as well as the dimensions of the R-Z

rectangle.

Integer vector of species numbers. This vector provides the species number to be plotted in the Mth frame.
  ISPEC=1,2,3,4,...,MAXSP,
Real vector of minimum R coordinate to be plotted for the Mth frame, i.e. RMIN(I).
  RMIN=0.0,0.0,...,
Real vector of maximum R coordinate to be plotted for the Mth frame, i.e. RMAX(I).
  RMAX=1.0,1.0,...,
Real vector of minimum Z coordinate to be plotted for the Mth frame, i.e. ZMIN(I).
  ZMIN=0.0,0.0,...,
Real vector of maximum Z coordinate to be plotted for the Mth frame, i.e. ZMAX(I).
  ZMAX=1.0,1.0,...,
R coordinate of viewpoint
  If ITIME=T, the value of RVIEW is actually along the time axis and the value of ZVIEW is actually along the spatial direction axis.
  RVIEW=100.0,
Z coordinate of viewpoint
  ZVIEW=100.0,
Function coordinate of viewpoint
  FVIEW=100.0,
Lower bound of the function axis
  FMATMN=0.0,
Upper bound on the function axis
  FMATMX=1.0,
Integer variable for the number of grid points used for the R coordinate direction in surface plotting.
  NRESR=NRESIN, (This variable defaults to the value given to NRESIN in Namelist FORMAT)
Integer variable for the number of grid points used for the Z coordinate direction in surface plotting.
  NRESZ=NRESIN, (This variable defaults to the value given to NRESIN in Namelist FORMAT)
  &END

6. Notes to above

Note 1. MATL has indices
(I=1 to NTIR+1,J=1 to NTIZ+1)

Note 2. HVGAP has indices
(KSPEC=1 to NSPEC,IGAP,J=1 to NTIZ+1)
where IGAP=1 to num. of vert. gaps along R.
HHGAP has indices
(KSPEC=1 to NSPEC,JGAP,I=1 to NTIR+1)
where JGAP=1 to num. of horz. gaps along Z.

Note 3. CK has indices
(KP=1 to NSPEC,K=1 to NSPEC)
CK(KP,K) is the first order reaction coefficient from the KP-th species into the K-th species. The terms CK(KP,K)*U(KP) are summed over kp to form the total first order reaction source into the K-th species.

Note 4. CKK has indices
(KPP=1 to NSPEC,KP=1 to NSPEC,K=1 to NSPEC)
CKK(KPP,KP,K) is the second order reaction coefficient from the KPP-th and the KP-th species into the K-th species. The terms CKK(KPP,KP,K)*U(KPP)*U(KP) are summed over KPP and KP to form the total second order reaction source for the K-th species.

Note 5. ALPHA, BETA, GAMMA have indices
(KSPEC=1 to NSPEC,I=1 to 4 sides)

Note 6. HU1 and HU3 have indices
(J=1 to NTIZ+1,KSPEC=1 to NSPEC)
HU2 and HU4 have indices
(I= 1 to NTIR+1,KSPEC=1 to NSPEC)

Note 7. NUTOUT: number of output time values including initial and final values
UTOUT: array of output times of dimension NUTOUT
NUFREQ: number of subintervals of time on the UTOUT time grid (PROUT is called at each subinterval time)

Note 8. SUBROUTINE ANAL(KSPEC,R,Z,VAL) must always be given.

Note 9. The graphics analytic subroutine has the form

```
SUBROUTINE ANAL(KSPEC,T,R,Z,V)
```

where KSPEC is the species number
T is the time
R and Z are the coordinates of the point to be evaluated
V is the value to be returned by the subroutine

A dummy version of this subroutine exists in the CSP code and thus the routine need only be provided if IANAL is TRUE.

Note 10. The use of namelist CSPIN is illustrated by the following examples (we assume IFORMT=1,i.e. packed format)

```
&CSPIN
 ISPEC=1,2,LGROUP=1,1,LORDER=1,2,
&END
```

Here we plot two species on a single frame. The first species is plotted on the first frame using the first calcomp symbol and the second species is also plotted on the first frame using the second plotting symbol. In the following example we plot three species with species 1 and 2 on the first frame and species 2 and 3 on the second frame. We use the Ith calcomp symbol to denote the plot of the Ith species.

```
&CSPIN
 ISPEC=1,2,2,3,,LGROUP=1,1,2,2,LORDER=1,2,2,3,
&END
```

In particular, the first components cause the first

species to be plotted on the first frame using the first calcomp plotting symbol. The second components cause the second species to be plotted on the first frame using the second plotting symbol. The third components cause the second species to be plotted on the second frame using the second plotting symbol. Finally, the fourth components cause the third species to be plotted on the second frame using the third calcomp plotting symbol.

APPENDIX B

List of Symbols

Section 1

Page

$\vec{n}$ . . . . . .Exterior unit normal to ∂R 14

h^g. . . . . .Gap interface coefficient. 15

u_m^0. . . . . .Initial distribution 16

Section 2

F . . . . . .General term for reaction and distributed source 18

R_s. . . . . .Material subrectangles 18

ω . . . . . .Test function in Galerkin procedure. 18

$C^1(R_s)$. . . .Class of functions continuous together with the first derivatives in R_s. 18

$\langle u,\omega \rangle$ . . . .Inner product of u,ω 18

$\partial\tilde{R}_1$ . . . . .That part of ∂R where $\beta \neq 0$. 18

$\partial\tilde{R}_2$ . . . . .That part of ∂R where no boundary condition is imposed . . . 18

∂R_0 . . . . .That part of ∂R where $\beta = 0$. 18

∂R_s . . . . .Boundary of R_s 18

Γ_s. . . . . .That part of ∂R_s which is an interface 19

π . . . . . .Mesh subdividing an interval 22

k . . . . . .Order of a polynomial. 22

$\mathbb{P}_{k,\pi}$. . . . .Space of piecewise polynomials 22

ℓ . . . . . .Number of subintervals in 22

X_i. . . . . .Breakpoints in π 22

ν_i. . . . . .Smoothness index associated with X_i. 22

$S_{k,\pi,\nu}$. . . .Spline subspace. 22

$g_k(\sigma;s)$ . . .Truncated power function 23

$\{\xi_i\}$. . . . .Knots. 23

$M_i(s)$ . . . .Divided difference of g_k 23

d_j. . . . . .Multiplicity of knots associated with X_j 23

Page

Section 4

ACKNOWLEDGMENTS

The authors wish to express their gratitude to T. Chawla for providing the motivation for undertaking this effort. We also express our appreciation to J. Dongarra and A. Hinds for their help and guidance in optimizing some key loops in this program. Special thanks to J. Beumer for the excellent typing of this report.

REFERENCES

[1] Hindmarsh, A. C., *GEAR: Ordinary Differential Equation System Solver*, Lawrence Livermore Laboratory Report UCID-30001, Rev. 2, Aug. 1972.

[2] De Boor, C., *Package for Calculating with B-Splines*, MRC Technical Summary Report No. 1333, Oct. 1973.

[3] De Boor, C., *On Calculating with B-Splines*, J. of Approximation Theory 6, 50-62 (1972).

[4] Strang, G. and Fix, G., *An Analysis of the Finite Element Method*, Prentice-Hall, Englewood Cliffs, NJ, 1973.

[5] Carslaw, H. S. and Jaeger, J. C., *Conduction of Heat in Solids*, Oxford University Press, 1948.

[6] Streeter, V. L. and Wylie, E. B., *Hydraulic Transients*, McGraw-Hill, New York, 1967.

[7] Byrne, G. D. and Hindmarsh, A. C., *EPISODEB: An Experimental Package for the Integration of Systems of Ordinary Differential Equations with Banded Jacobians*, Lawrence Livermore Laboratory Report UCID-30132, April 30, 1976.

[8] Bird, R. B., Stewart, W. E., and Lightfoot, E. N., *Transport Phenomena*, Wiley & Sons, New York, 1960.

[9] Prastein, S. M., ed., *Argonne National Laboratory User's Guide*, June 1976.

[10] CALCOMP, *Programming Calcomp Electromechanical Plotters*, January, 1976, California Computer Products, Inc., Anaheim, CA.

[11] Malon, D. M., *CØNTØUR.BLACKBØX, A Contour Mapping Program*, Applied Mathematics Division Technical Memo No. 277, Argonne National Laboratory, Oct. 1, 1975.

[12] DISSPLA, *Display Integrated Software System and Plotting Language, Vols. I and II*, Integrated Software Systems Corp., 5th printing, July, 1975.

[13] IBM, *IBM System/360 and System/370 FORTRAN IV Language*, Order No. GC28-6515-10, Eleventh Edition, May, 1974, New York, New York.

[14] Cook, A. J. and Shustek, L. J., *A User's Guide to MORTRAN*, Computation Research Group, Stanford Linear Accelerator Center, CGTM No. 165, Stanford, California, 1975.

Distribution for ANL-77-12, Rev. 1

Internal:

M. V. Nevitt
B. Ancker-Johnson
R. J. Royston
G. K. Leaf (73)
D. M. Pahis (10)
A. B. Krisciunas
ANL Contract File
ANL Libraries (5)
TIS Files (6)

External:

DOE-TIC, for distribution per UC-32 (203)
Manager, Chicago Operations Office
Chief, Office of Patent Counsel, CH
President, Argonne Universities Association
J. C. T. Pool, BES, USDOE
Applied Mathematics Division Review Committee:
  P. J. Eberlein, SUNY at Buffalo
  G. Estrin, U. California, Los Angeles
  W. M. Gentleman, U. Waterloo
  J. M. Ortega, NASA Langley Research Center
  E. N. Pinson, Bell Telephone Labs.
  S. Rosen, Purdue U.
  D. M. Young, Jr., U. Texas, Austin